

(19) 世界知的所有権機関  
国際事務局(43) 国際公開日  
2005 年 8 月 4 日 (04.08.2005)

PCT

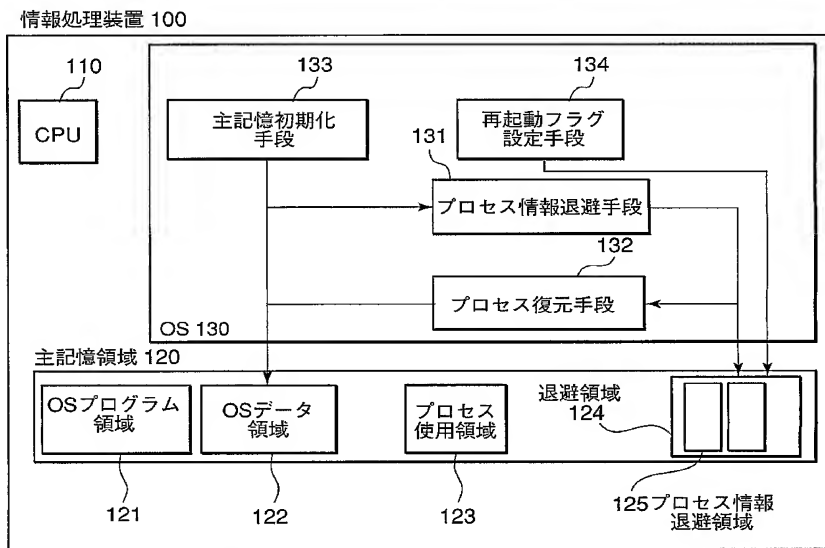
(10) 国際公開番号  
WO 2005/071522 A1

- (51) 国際特許分類: G06F 1/24, 9/445, 9/46  
(21) 国際出願番号: PCT/JP2005/001559  
(22) 国際出願日: 2005 年 1 月 27 日 (27.01.2005)  
(25) 国際出願の言語: 日本語  
(26) 国際公開の言語: 日本語  
(30) 優先権データ:  
特願2004-018524 2004 年 1 月 27 日 (27.01.2004) JP  
(71) 出願人 (米国を除く全ての指定国について): 日本電気株式会社 (NEC CORPORATION) [JP/JP]; 〒1088001 東京都港区芝五丁目 7 番 1 号 Tokyo (JP).  
(72) 発明者; および  
(75) 発明者/出願人 (米国についてのみ): 小比賀 亮仁 (KOHIGA, Akihito) [JP/JP]; 〒1088001 東京都港区芝五丁目 7 番 1 号 日本電気株式会社内 Tokyo (JP).  
(74) 代理人: 池田 憲保 (IKEDA, Noriyasu); 〒1050003 東京都港区西新橋一丁目 4 番 1 0 号 第 3 森ビル Tokyo (JP).  
(81) 指定国 (表示のない限り、全ての種類の国内保護が可能): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE,

[続葉有]

(54) Title: HIGH-SPEED RESTART METHOD, INFORMATION PROCESSING DEVICE, AND PROGRAM

(54) 発明の名称: 高速再起動方法および情報処理装置ならびにプログラム



100... INFORMATION PROCESSING DEVICE  
133... MAIN STORAGE INITIALIZATION MEANS  
134... RESTART FLAG SETTING MEANS  
131... PROCESS INFORMATION SAVING MEANS  
132... PROCESS RESTORATION MEANS  
120... MAIN STORAGE AREA  
121... OS PROGRAM AREA  
122... OS DATA AREA  
123... PROCESS USE AREA  
124... SAVE AREA  
125... PROCESS INFORMATION SAVE AREA

(57) Abstract: An information processing device includes process information saving means (131) and process restoration means (132). During execution of an application, a process table (201) existing in an OS (130) and other information required for restoring the process are copied to a save area (124) on a main storage area (120). During restart, the main storage area used by the application is maintained as it is and only the OS is initialized. After the initialization, the process information (the process table and other information managed by the OS) which has been saved is overwritten on an OS data area (122). By using such a configuration, it is possible to provide an environment where the process before the restart can be continuously executed even after the process restart.

(57) 要約: プロセス情報退避手段131と、プロセス復元手段132とを備え、アプリケーションの実行時にOS 130内に存在するプロセス表201やその他、プロセスを復元するために必要な情報を主記憶領域120上の退避領域124に複写しておき、再起動時には、アプリケーションの使用

[続葉有]

WO 2005/071522 A1



SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US,  
UZ, VC, VN, YU, ZA, ZM, ZW.

OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML,  
MR, NE, SN, TD, TG).

(84) 指定国 (表示のない限り、全ての種類の広域保護  
が可能): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA,  
SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ,  
BY, KG, KZ, MD, RU, TJ, TM), ヨーロッパ (AT, BE,  
BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU,  
IE, IS, IT, LT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR),

添付公開書類:

— 国際調査報告書

2文字コード及び他の略語については、定期発行される  
各PCTガゼットの巻頭に掲載されている「コードと略語  
のガイダンスノート」を参照。

## 明 細 書

## 高速再起動方法および情報処理装置ならびにプログラム

## 技術分野

本発明は、AP サーバーや汎用情報処理装置などに好適な高速再起動方法に関し、特に、再起動前に運用していたアプリケーションをそのまま再起動後も継続して運用できる高速再起動方法および情報処理装置ならびにプログラムに関する。

## 背景技術

再起動は、OS の設定を変更した場合、設定を反映させるために利用される。Windows(登録商標)OS の場合は、レジストリの内容を変更し、変更を反映させるために再起動が必要になる。さらに、システムに何らかの障害が発生し、発生した障害の原因を特定することが困難である場合は、いち早く障害からシステムを回復させる最初的手段として、再起動を用いることが多い。このような再起動は、システムを運用している間は頻繁に利用するものであり、システムの停止時間を最小限にとどめるため、再起動処理は高速化が望まれていた。

再起動処理の高速化を実現するための従来の技術の例としては、特開平 11-24936 号公報第 7 頁、図 1 等に記載される情報処理装置の高速再起動方法がある。図 1 はこの従来の情報処理装置の構成を示したものである。図 1 に示すように従来の情報処理装置 1 は、その中に主記憶装置 11 が配置され、外部には入出力装置 2 がある。また、主記憶装置 11 の内部にはオペレーティングシステム(以下、OS と呼ぶ)がロードされる OS プログラム領域 111、初期化データ領域 112、非初期化データ領域 113、初期化データ退避領域 114 が配置されており、主記憶装置 11 以外の情報処理装置内部は、再起動開始手段 12、高速再起動指定手段 13、実行制御手段 14、初期化データ退避手段 15、初期化データ回復手段 16、入出力停止手段 17、OS ロード手段 18 から構成されている。

この従来の情報処理装置 1 では、OS 起動時に OS の初期化データを初期化データ

退避手段 15 により初期化データ退避領域 114 に退避し、高速再起動指定手段 13 により高速再起動が指定されると、実行制御手段 14 により OS プログラムの実行開始番地に実行を移し、初期化データ回復手段 16 により初期化データを回復する。以上の処理を行うことで、OS のプログラムおよびデータを再ロードすることなく情報処理装置を高速に再起動する高速再起動方式を提供している。

しかしながら、特許文献 1 に記載された従来の高速再起動方法では、OS の高速再起動を実現することはできるが、OS 上で実行されているアプリケーション(アプリケーションプログラム)については言及されていない。OS を再起動する際にはアプリケーションを正常終了または強制終了させる必要がある。再起動時間を短縮するためにアプリケーションを強制終了させると、再起動後の当該アプリケーションの正常運用を保障できない。なぜなら、アプリケーションの強制終了は、現在のサービス状態を保存しない。アプリケーションによっては、運用時のサービス状態が再起動後に必要な場合がある。そのような場合には再起動後にアプリケーションの正常運用を保障することができなくなる。よって、アプリケーションを正常終了させる必要があるが、アプリケーションの正常終了は、現在のサービス状態を保存するために、ディスクなどの処理速度の遅い補助記憶装置へのアクセスが頻繁に発生する。このような補助記憶装置へのアクセスは、システムの終了処理時間を増大させることとなり、システムの再起動時間を増大させるものとなる。

また、再起動を利用する際に考慮すべき事項として、次のような場合が想定される。再起動は主にシステムの設定を OS に反映させたり、あるアプリケーションの障害により OS が正常に動作しなくなった場合に利用される。複数のアプリケーションを使ってサービスを提供するサーバにおいて、ある一つのアプリケーションに障害が発生し、OS が正常に動作しなくなった場合は、再起動する必要があるが、その際には、障害の発生したアプリケーション以外のアプリケーションも終了させる必要があり、サービス途中にあるアプリケーションはサービスを強制中断せざるを得なくなる。

上記の問題点を克服するために、本発明では、シャットダウン時の処理を最小限にとどめて、システムを高速に再起動できる高速再起動方法および情報処理装置な

らびにプログラムを提供することを目的とする。

本発明の他の目的は、再起動後もアプリケーションを継続運用できる高速再起動方法および情報処理装置ならびにプログラムを提供することを目的とする。

## 発明の開示

課題を解決するための手段を説明するにあたり、本発明を理解するための要素技術の説明を行う。

計算機はプログラムを実行する。計算機中のプログラムはプロセスという実行単位に分割されて実行される。ある時点において計算機が実行できるプロセスは一つであるので、プロセスを複数同時に実行するには、当該プロセスを管理する上位プログラムが必要となる。それが OS にあたるが、OS は複数のプロセスを管理するために、OS 自身が利用している主記憶領域上にプロセス表というものを持っている。プロセス表はプロセス一つにつき一つ用意され、プロセスを実行するための情報や、実行途中の情報を格納している。よって、あるプロセスは当該プロセスの利用している主記憶領域とプロセス表を保存しておくことにより、実行途中にあるプロセスを中断したり、中断した時点から処理を再開することができる。一つまたは複数のプロセスによってアプリケーションが構成される。そこで本発明では、アプリケーションの実行時などに OS 内に存在するプロセス表やその他、プロセスを復元するために必要な情報を主記憶領域上の退避領域に複写しておき、再起動時には、アプリケーションの使用していた主記憶の領域はそのままに、OS のみを初期化する。そして初期化の後に、退避しておいたプロセス情報(プロセス表やその他 OS が管理していた情報)を OS のプロセス管理領域に上書きするよう動作する。このような構成を採用することで、再起動前のプロセスを再起動後も継続実行できる環境を提供することができる。より具体的には、本発明は以下のような構成を有する。

本発明の第 1 の高速再起動方法は、OS の再起動前に、ユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させ、OS の再起動時に、前記ユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化し、OS 再起動後に、前記退避したプロセス情報を OS 内に復元す

ることを特徴とする。

本発明の第 2 の高速再起動方法は、OS の再起動前に、ユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させると共に、前記退避させたプロセス情報に再起動するかどうかを指定する再起動フラグを設定し、OS の再起動時に、前記再起動フラグが再起動しない設定になっているユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化し、OS 再起動後に、前記退避したプロセス情報のうち前記再起動フラグが再起動しない設定になっているユーザプロセスのプロセス情報を OS 内に復元することを特徴とする。

本発明の第 3 の高速再起動方法は、OS の再起動前に、OS 再起動後に継続運用するユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させ、OS の再起動時に、前記ユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化し、OS 再起動後に、前記退避したプロセス情報を OS 内に復元することを特徴とする。ここで、OS 再起動後に継続運用するユーザプロセスかどうか、あるいはその反対に OS 再起動時に再起動するユーザプロセスかどうかは、例えば、OS の再起動後に継続運用するプロセスまたは継続運用しないプロセスの識別子を保存してあるプロセス ID テーブルを参照して、判断することができる。

本発明の第 4 の高速再起動方法は、ユーザプロセスの生成時、前記生成したユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させ、前記ユーザプロセスの切替え時、前記退避させたプロセス情報に再起動するかどうかを指定する再起動フラグを設定するとともに、再起動しない場合には退避領域に退避されているプロセス情報を最新の状態に更新し、前記ユーザプロセスの終了時、前記退避させたプロセス情報を無効にし、OS の再起動時に、前記再起動フラグが再起動しない設定になっているユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化し、OS 再起動後に、前記退避したプロセス情報のうち前記再起動フラグが再起動しない設定になっているユーザプロセスのプロセス情報を OS 内に復元することを特徴とする。

本発明の第 5 の高速再起動方法は、第 2 または第 4 の高速再起動方法において、或るユーザプロセスに関するプロセス情報に再起動するかどうかを指定する再起動フラグを設定する際、そのユーザプロセスと同じユーザアプリケーションプログラムに属する他の全てのユーザプロセスを検索し、該検索した全てのユーザプロセスに関するプロセス情報中の再起動フラグも同じ値に設定することを特徴とする。

本発明の第 6 の高速再起動方法は、第 1 乃至第 4 の何れかの高速再起動方法において、OS の使用していた主記憶領域の初期化は、主記憶装置の一部を構成する不揮発性記憶部に記憶された OS の主記憶イメージを OS の使用していた主記憶領域に上書きすることで行うことを特徴とする。

本発明の第 7 の高速再起動方法は、第 6 の高速再起動方法において、システム運用中に OS から前記不揮発性記憶部への書き込みアクセスが発生する毎に、書き込みアクセスが発生したアドレスを含む所定幅のアドレス範囲内のデータを前記不揮発性記憶部から主記憶装置の一部を構成する読み出し書き込み可能な主記憶部に確保した代替領域にコピーし且つ以後の前記所定幅のアドレス範囲へのアクセスを前記代替領域へのアクセスに変換することを特徴とする。

本発明の第 1 の情報処理装置は、OS の再起動前に、ユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス情報退避手段と、OS の再起動時に、前記ユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段と、OS 再起動後に、前記退避したプロセス情報を OS 内に復元するプロセス復元手段とを備えることを特徴とする。

本発明の第 2 の情報処理装置は、OS の再起動前に、ユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス情報退避手段と、前記退避させたプロセス情報に再起動するかどうかを指定する再起動フラグを設定する再起動フラグ設定手段と、OS の再起動時に、前記再起動フラグが再起動しない設定になっているユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段と、OS 再起動後に、前記退避したプロセス情報のうち前記再起動フラグが再起動しない設定になって

いるユーザプロセスのプロセス情報を OS 内に復元するプロセス復元手段とを備えることを特徴とする。

本発明の第 3 の情報処理装置は、OS の再起動前に、OS 再起動後に継続運用するユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス情報退避手段と、OS の再起動時に、前記ユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段と、OS 再起動後に、前記退避したプロセス情報を OS 内に復元するプロセス復元手段とを備えることを特徴とする。ここで、プロセス情報退避手段は、例えば、OS の再起動前に、OS の再起動後に継続運用するプロセスまたは継続運用しないプロセスの識別子を保存するプロセス ID テーブルを参照して、OS 再起動後に継続運用するユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるものである。

本発明の第 4 の情報処理装置は、ユーザプロセスの生成時、前記生成したユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス退避領域作成手段と、前記ユーザプロセスの切替え時、前記退避させたプロセス情報に再起動するかどうかを指定する再起動フラグを設定するとともに、再起動しない場合には退避領域に退避されているプロセス情報を最新の状態に更新するプロセス退避情報更新手段と、前記ユーザプロセスの終了時、前記退避させたプロセス情報を無効にするプロセス情報退避領域解放手段と、OS の再起動時に、前記再起動フラグが再起動しない設定になっているユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段と、OS 再起動後に、前記退避したプロセス情報のうち前記再起動フラグが再起動しない設定になっているユーザプロセスのプロセス情報を OS 内に復元するプロセス復元手段とを備えることを特徴とする。

本発明の第 5 の情報処理装置は、第 2 または第 4 の情報処理装置において、或るユーザプロセスに関するプロセス情報に再起動するかどうかを指定する再起動フラグを設定する際、そのユーザプロセスと同じユーザアプリケーションプログラムに属する他の全てのユーザプロセスを検索し、該検索した全てのユーザプロセスに

関するプロセス情報中の再起動フラグも同じ値に設定する手段を備えることを特徴とする。

本発明の第 6 の情報処理装置は、第 1 乃至第 4 の何れかの情報処理装置において、前記主記憶初期化手段は、OS の使用していた主記憶領域の初期化は、主記憶装置の一部を構成する不揮発性記憶部に記憶された OS の主記憶イメージを OS の使用していた主記憶領域に上書きすることで行うものであることを特徴とする。

本発明の第 7 の情報処理装置は、第 6 の情報処理装置において、システム運用中に OS から前記不揮発性記憶部への書き込みアクセスが発生する毎に、書き込みアクセスが発生したアドレスを含む所定幅のアドレス範囲内のデータを前記不揮発性記憶部から主記憶装置の一部を構成する読み出し書き込み可能な主記憶部に確保した代替領域にコピーし且つ以後の前記所定幅のアドレス範囲へのアクセスを前記代替領域へのアクセスに変換する手段を備えることを特徴とする。

本発明の第 1 のプログラムは、コンピュータを、OS の再起動前に、ユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス情報退避手段、OS の再起動時に、前記ユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段、OS 再起動後に、前記退避したプロセス情報を OS 内に復元するプロセス復元手段、として機能させることを特徴とする。

本発明の第 2 のプログラムは、コンピュータを、OS の再起動前に、ユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス情報退避手段、前記退避させたプロセス情報に再起動するかどうかを指定する再起動フラグを設定する再起動フラグ設定手段、OS の再起動時に、前記再起動フラグが再起動しない設定になっているユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段、OS 再起動後に、前記退避したプロセス情報のうち前記再起動フラグが再起動しない設定になっているユーザプロセスのプロセス情報を OS 内に復元するプロセス復元手段、として機能させることを特徴とする。

本発明の第 3 のプログラムは、コンピュータを、OS の再起動前に、OS 再起動後

に継続運用するユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス情報退避手段、OS の再起動時に、前記ユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段、OS 再起動後に、前記退避したプロセス情報を OS 内に復元するプロセス復元手段、として機能させることを特徴とする。ここで、プロセス情報退避手段は、例えば、OS の再起動前に、OS の再起動後に継続運用するプロセスまたは継続運用しないプロセスの識別子を保存するプロセス ID テーブルを参照して、OS 再起動後に継続運用するユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるものである。

本発明の第 4 のプログラムは、コンピュータを、ユーザプロセスの生成時、前記生成したユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス退避領域作成手段、前記ユーザプロセスの切替え時、前記退避させたプロセス情報に再起動するかどうかを指定する再起動フラグを設定するとともに、再起動しない場合には退避領域に退避されているプロセス情報を最新の状態に更新するプロセス退避情報更新手段、前記ユーザプロセスの終了時、前記退避させたプロセス情報を無効にするプロセス情報退避領域解放手段、OS の再起動時に、前記再起動フラグが再起動しない設定になっているユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段、OS 再起動後に、前記退避したプロセス情報のうち前記再起動フラグが再起動しない設定になっているユーザプロセスのプロセス情報を OS 内に復元するプロセス復元手段、として機能させることを特徴とする。

本発明の第 5 のプログラムは、第 2 または第 4 のプログラムにおいて、コンピュータを、さらに、或るユーザプロセスに関するプロセス情報に再起動するかどうかを指定する再起動フラグを設定する際、そのユーザプロセスと同じユーザアプリケーションプログラムに属する他の全てのユーザプロセスを検索し、該検索した全てのユーザプロセスに関するプロセス情報中の再起動フラグも同じ値に設定する手段、として機能させることを特徴とする。

本発明の第 6 のプログラムは、第 1 乃至第 4 の何れかのプログラムにおいて、前

記主記憶初期化手段は、OS の使用していた主記憶領域の初期化は、主記憶装置の一部を構成する不揮発性記憶部に記憶された OS の主記憶イメージを OS の使用していた主記憶領域に上書きすることで行うものであることを特徴とする。

本発明の第 7 のプログラムは、第 6 のプログラムにおいて、コンピュータを、さらに、システム運用中に OS から前記不揮発性記憶部への書き込みアクセスが発生する毎に、書き込みアクセスが発生したアドレスを含む所定幅のアドレス範囲内のデータを前記不揮発性記憶部から主記憶装置の一部を構成する読み出し書き込み可能な主記憶部に確保した代替領域にコピーし且つ以後の前記所定幅のアドレス範囲へのアクセスを前記代替領域へのアクセスに変換する手段、として機能させることを特徴とする。

更に、本発明は、次のような OS 再起動方法を提供する。即ち、メインメモリ上に、OS をロードするためのメモリ領域である第 1 の OS メモリ領域と、プロセスをロードするためのメモリ領域であるプロセスメモリ領域とを割り当てて、OS 及びプロセスを該当する領域にロードしたコンピュータで、OS を再起動する方法において、OS がプロセスを管理するための情報であるプロセス情報を、第 1 の OS メモリ領域から取得し、予め定められた記憶装置に設けた退避領域に格納する段階 1 と、プロセスメモリ領域を保持したまま、第 1 の OS メモリ領域を初期化する段階 2 と、メインメモリ上に第 2 の OS メモリ領域を割り当てて、OS をロードする段階 3 と、OS メモリ領域中のプロセス情報を、段階 1 にて格納されたプロセス情報に応じて更新する段階 4 とを含むことを特徴とする OS 再起動方法を提供する。

この OS 再起動方法において、更に、プロセスメモリ領域にロードされたプロセスから保持すべきプロセスを選択する段階と、選択されなかったプロセスに割り当てられたプロセスメモリ領域を初期化する段階とを含むこととしてもよい。

この OS 再起動方法において、退避領域をメインメモリ上に設けることとしてもよい。

この OS 再起動方法において、各プロセスを再起動するか否かを示す情報を、当該プロセスのプロセス情報と共に、退避領域に記憶することとしてもよい。

この OS 再起動方法において、各プロセスを再起動するか否かを示す情報を、退

避領域を設けた記憶装置とは別の記憶装置に記憶することとしてもよい。

このOS再起動方法において、プロセスの生成、切り替え及び終了に応じて、退避領域を生成、更新及び解放する処理を、退避領域を設けた記憶装置に対して実行することとしてもよい。

このOS再起動方法において、OSをメインメモリにロードしたときのイメージを格納した不揮発性記憶装置を予め用意する方法であって、段階3は、不揮発性記憶装置に格納したイメージを参照してメインメモリにOSをロードすることとしてもよい。

このOS再起動方法において、更に、一のアプリケーションプログラムに対して関連付けられた複数のプロセスを含むプロセスを、プロセスメモリ領域にロードする段階と、プロセスメモリ領域にロードされたプロセスの中から、保持すべきプロセスを選択する段階と、選択されたプロセス、及び、当該プロセスと同じアプリケーションプログラムに関連付けられた他のプロセス以外のプロセスに割り当てられたプロセスメモリ領域を、初期化する段階とを含むこととしてもよい。

#### 図面の簡単な説明

図1は従来の情報処理装置の構成を示すブロック図であり、

図2は本発明の第1の実施の形態にかかる情報処理装置の構成を示すブロック図であり、

図3は本発明の第1の実施の形態における退避領域のデータ構造を示す図であり、

図4A及び図4Bは本発明の第1の実施の形態の動作を示す流れ図であり、

図5は本発明の第1の実施の形態におけるブロック識別子を示す図であり、

図6は本発明の第1の実施の形態における主記憶初期化手段の処理を示す流れ図であり、

図7は本発明の第1の実施の形態の変形例の構成を示すブロック図であり、

図8は本発明の第1の実施の形態の変形例の動作を示す流れ図であり、

図9は本発明の第2の実施の形態にかかる情報処理装置の構成を示すブロック

図であり、

図 1 0 は本発明の第 2 の実施の形態におけるプロセス退避情報作成手段の処理を示す流れ図であり、

図 1 1 は本発明の第 2 の実施の形態におけるプロセス退避情報更新手段の処理を示す流れ図であり、

図 1 2 は本発明の第 2 の実施の形態におけるプロセス情報退避領域解放手段の処理を示す流れ図であり、

図 1 3 は本発明の第 3 の実施の形態で利用する主記憶イメージを用いた高速起動方式を適用した情報処理装置の構成図であり、

図 1 4 は本発明の第 3 の実施の形態で利用する主記憶イメージを用いた高速起動方式を適用した情報処理装置におけるマッピングテーブル部のエントリ構成を示す図であり、

図 1 5 は本発明の第 3 の実施の形態で利用する主記憶イメージを用いた高速起動方式を適用した情報処理装置における保護情報部のエントリ構成を示す図であり、

図 1 6 は本発明の第 3 の実施の形態で利用する主記憶イメージを用いた高速起動方式を適用した情報処理装置の処理を示す流れ図であり、

図 1 7 は本発明の第 3 の実施の形態で利用する主記憶イメージを用いた高速起動方式を適用した情報処理装置の処理を示す流れ図であり、

図 1 8 は本発明の第 3 の実施の形態にかかる情報処理装置の構成を示すブロック図であり、

図 1 9 は本発明の第 3 の実施の形態にかかる情報処理装置における主記憶初期化手段の処理を示す流れ図であり、

図 2 0 は本発明の第 4 の実施の形態にかかる情報処理装置の構成を示すブロック図であり、

図 2 1 は本発明の第 4 の実施の形態にかかる情報処理装置における再起動フラグ設定手段の処理を示す流れ図であり、

図 2 2 は本発明の第 1 の実施例におけるプロセス情報退避手段の処理を示す流

れ図であり、

図 2 3 は本発明の第 1 の実施例における主記憶初期化手段の処理を示す流れ図であり、

図 2 4 は本発明の第 2 の実施例におけるプロセス退避情報作成手段の処理を示す流れ図であり、

図 2 5 は本発明の第 2 の実施例におけるプロセス退避情報更新手段の処理を示す流れ図である。

発明を実施するための最良の形態

次に、本発明の第 1 の実施の形態について図面を参照して詳細に説明する。

図 2 を参照すると、本発明の第 1 の実施の形態にかかる情報処理装置 100 は、プログラム制御により動作する CPU(プロセッサ)110 と、主記憶領域 120 と、OS130 から構成されている。プロセスは CPU110 により実行される。主記憶領域 120 は、OS プログラム領域 121、OS データ領域 122、プロセス使用領域 123、退避領域 124 を含み、退避領域 124 内には実行されているプロセス分だけプロセス情報退避領域 125 が存在する。OS130 は、プロセス情報退避手段 131、プロセス復元手段 132、主記憶初期化手段 133、再起動フラグ設定手段 134 を含む。

図 3 に退避領域 124 のデータ構造を示す。退避領域 124 には、プロセス情報退避領域 125 とよばれる個々のプロセスごとに OS の管理している情報を退避させる領域があり、プロセス情報退避領域 125 は単方向リスト 205 でつながれている。さらに、退避領域使用サイズ 201 という退避領域 124 の大きさを示すデータ領域を持つ。OS の管理している情報とは、プロセス表 202 とブロックテーブル 203 である。OS は 1 つのプロセスにつき 1 つのプロセス表を作成して保守している。プロセス表には、プロセスの状態、プログラムのカウンタやスタックポインタ、メモリの割り当て、オープンしているファイルの状態やスケジュール情報、プロセスに関するあらゆる情報が入っている。また、OS は主記憶領域 120 をブロックという単位で管理しており、ブロックテーブル 203 とは、個々のプロセスに一つ存在し、当該プロセスの使用しているブロックのアドレスが記述されている。再起動フラグ 204 は、当

該プロセスを再起動させるかどうかを判断するためのフラグである。本実施の形態では、プロセス表 202 とは独立に再起動フラグ 204 を設けてあるが、プロセス表 202 の中に再起動フラグを持つようにしてもよい。

退避領域 124 は、基本的に主記憶領域 120 の一番後方のアドレスから始まり、主記憶領域 120 の先頭に向かって、現在実行しているプロセスの数だけ主記憶領域 120 を使用する。退避領域 124 を後方の定位置からアドレッシングする方法は、次のような利点がある。OS は、起動の際に主記憶領域の管理を行うために、主記憶領域の初期化を行う。本発明において主記憶領域初期化の際に OS は、再起動前に使用されていた退避領域 124 と、プロセス自身がプログラムを実行するために使用していたプロセス使用領域 123 を知る必要がある。再起動前にプロセスによって使用されていたプロセス使用領域 123 を起動後に知るには、定位置から退避領域 124 を置くとよい。起動後に OS は、退避領域 124 がどこにあるのかを調べる必要がなく、OS プログラムの一部として「プロセス情報退避領域 125 は主記憶領域 120 の一番後方のアドレスから置かれる」と記述することで、プロセス情報退避領域 125 がどこにあるのかを知ることができ、退避領域 124 のプロセス情報退避領域 125 からプロセス使用領域 123 を知ることができる。よって、再起動前の退避領域 124 の位置がわかりさえすれば退避領域 124 の主記憶領域 120 上での位置は一番後方でなくてもよく、また別の記憶装置(フラッシュメモリなどの不揮発性主記憶装置)に退避領域 124 を保存しておいてもよい。

プロセス情報退避手段 131、プロセス復元手段 132、主記憶初期化手段 133、再起動フラグ設定手段 134 は、それぞれ次のような機能を有する。

プロセス情報退避手段 131 は、OS 内部にあるプロセス表とブロックテーブルをプロセス情報退避領域 125 に退避させる。

プロセス表の退避処理とは、あるプロセスについて、プロセス表をプロセス情報退避領域 125 へ OS データ領域 122 からコピーすることである。

通常、OS は主記憶領域 120 を複数のブロックに分割して管理しており、OS はプロセスに必要な量のブロックを与えることで、プロセスを実行している。どのブロックを当該プロセスが使用しているかという情報はブロックテーブルに保存されて

いる。ブロックテーブルの退避処理とは、プロセスに付随するブロックテーブル 203 を退避領域 124 に OS データ領域 122 からコピーすることである。

再起動フラグ設定手段 134 は、ユーザからの入力によって、あるプロセス情報退避領域 125 内に設けられた再起動フラグ 204 を設定する手段である。ユーザからの入力は、プロセスを特定するためのプロセス ID である。再起動フラグ 204 を設定されたプロセスは、再起動されるため、再起動後に再起動前の状態に復元せず、当該プロセスのプロセス情報退避領域 125 と、プロセス使用領域 123 は解放される。これに対して、再起動フラグ 204 を設定されないプロセスは、再起動されず、当該プロセスのプロセス情報退避領域 125 とプロセス使用領域 123 を用いて、再起動前の状態に復元される。

主記憶初期化手段 133 は、OS が起動前に実行していたプロセスについて、当該プロセスが利用していた領域を認識するための処理である。通常、OS は起動時に利用できる主記憶領域 120 を認識するために、ブロックごとに初期化処理を行う。ここでプロセス情報退避領域 125 より、起動前に実行していたプロセスが使用しているブロックをブロックテーブル 203 より検索し、使用しているブロックについては初期化処理を施さないようにする。

プロセス復元手段 132 は、プロセス情報退避領域 125 に格納されているプロセス表 202、ブロックテーブル 203 からプロセスを復元する処理である。復元処理とは、OS データ領域 122 にプロセス表 202 およびブロックテーブル 203 を書き戻す(コピー)処理である。

次に、図 2 ないし図 6 を参照して本実施の形態について全体の動作を詳細に説明する。

最初に、システム終了時までの処理について説明する。まず、全てのプロセスについて、プロセス情報退避手段 131 を用いてプロセス情報退避領域 125 を作成する(図 4 A S100)。この処理は全てのプロセスについて行い、全てのプロセスについてプロセス情報退避領域 125 が作成できたとき終了する(S101)。

プロセス情報退避領域 125 の作成は、まず主記憶上にプロセス情報退避領域 125 を確保し、当該プロセスのプロセス表 202 を OS データ領域 122 から読み出して、

確保したプロセス情報退避領域 125 にコピーし、次に OS データ領域 122 からブロックテーブル 203 を読み出して、確保したプロセス情報退避領域 125 にコピーすることで行う。また、作成時には退避領域使用サイズ 201 を当該プロセスのために使用した主記憶領域分だけ足す。さらに、必要に応じて再起動フラグ 204 を設定し、単方向リスト 205 でプロセス情報退避領域 125 どうしをつなげる。

こうして全てのプロセスについてプロセス情報退避領域 125 の作成を行った後、OS の通常終了処理を行う (S102)。プロセスについては終了処理をせず、そのままの状態でも OS 起動後のプロセスの復元処理を待つ。プロセス情報退避領域 125 の作成は、システム終了直前、つまりユーザがシステム終了命令を発行してから作成を開始し、OS の終了処理を始めるまでの間に作成するが、任意の時間に作成してもよい。

再起動フラグ設定手段 134 による再起動フラグ 204 の設定は、システム終了までの任意の時間に行う。再起動フラグ 204 は、プロセス情報退避領域 125 の中に設置されており、当該プロセスの退避領域が作成されるまで設定することができないので、当該プロセスの退避領域が作成される前にユーザから再起動する旨の指示があれば、OS は一時的に当該プロセスに対する再起動フラグ設定の情報を保持しておく必要がある。その際の設定の情報は、再起動フラグ設定手段 134 の中にプロセス ID のリストとして保持しておく。その後プロセス情報退避領域 125 が作成されたら、以前のユーザの指示に従い、再起動フラグ 204 を設定する。

次に、起動時の処理を説明する。OS の展開を含む OS の起動 (S103) については、Linux などの OS と同様である。Linux の展開については、例えば参考文献 1「D. Bovet and M. Cesati. 詳解 Linux カーネル, オライリー・ジャパン, 2001.」に記載されている。起動時の処理の説明は、OS の主記憶上への展開が終了し、OS が最初の処理を行うところから始める。

まず、主記憶領域 120 の利用可能領域を認識するために OS により主記憶の初期化がブロック単位で行われる (S104)。この初期化の際には再起動前にプロセスによって使用されていた領域を OS に認識させる必要がある。OS はあるブロックが使用中であるか否かを認識するために、ブロック識別子 400 を持つ (図 5)。ブロック識

別子 400 は、ブロック番号 401 と使用フラグ 402 とから構成される。ブロック識別子 400 は一つのブロックについて一つ存在する。起動時には、OS データ領域 122 にブロック識別子 400 のための領域を確保し、使用フラグ 402 を零(使用されていない)に設定することにより主記憶領域 120 の初期化とみなされる。利用可能である主記憶領域の開始アドレスは、あらかじめ BIOS などによってその値が与えられている。

図 6 は本発明における主記憶初期化手段 133 のフローチャートを示している。本発明における主記憶初期化手段 133 を実行する前にあらかじめ退避領域 124 の使用している領域を確認する(S201)。退避領域 124 の使用している領域は、退避領域 124 にある退避領域使用サイズ 201 を見ることで確認することができる。次に、退避領域 124 内にあるすべてのプロセス情報退避領域 125 を検索し、再起動フラグ 204 の設定されていない(再起動前のプロセスを継続利用する)プロセスのブロックテーブル 203 からあらかじめ初期化の必要のないブロックを把握しておく(S202)。退避領域 124 によって使用されているブロックのブロック番号 401 のリストを一時的に保持しておき、のちの処理に使用する。

退避領域使用サイズ 201 を確認し、初期化の必要のないブロックを把握した上で、指定されたアドレスに対するブロック識別子一つ分の主記憶領域を 0 で初期化する(S204)。次に、退避領域 124 で先ほど初期化したブロックが使用されているかを確認する(S205)。使用しているなら(S206 の YES)、使用フラグを立て(S207)、次のアドレスへ移る。このようにして、利用可能である領域の全てのブロック識別子 400 を初期化すると(S203 の YES)主記憶初期化手段 133 を終了する。

主記憶領域 120 の初期化が終了すると、次にプロセスを復元する。退避領域 124 内のプロセス情報退避領域 125 を確認し、再起動フラグ 204 が設定されているか確認する(S107)。再起動を行うのであれば(S108)、当該プロセス情報の退避領域 124 を解放し、次のプロセスの退避情報に関して処理を行う(S105)。再起動を行わないのであれば、プロセスを復元する(S109)。以上の処理を退避領域 124 内にあるプロセス情報退避領域 125 のエントリが全てなくなるまで行う(S106)。プロセスの復元は、プロセス表とブロックテーブルのための領域を OS データ領域 122 上に確保し

(S110、S112)、退避しておいたプロセス表 202 とブロックテーブル 203 を用意した領域にコピーする(S111、S113)。あるプロセス情報の退避領域 124 の解放時には、以下の処理を行う。

1) 削除するプロセス情報退避領域の一つ前のプロセス情報退避領域(単方向リスト 205 での一つ前のリスト要素)の単方向リスト 205 が指す次のプロセス情報退避領域へのポインタを、削除するプロセス情報退避領域に格納されている単方向リスト 205 の内容に置き換える。

2) 削除するプロセス情報退避領域が使用している主記憶ブロックのブロック識別子 400 内の使用フラグ 402 を零(使用していない)にする。

3) 退避領域使用サイズ 201 の値から、削除するプロセス情報退避領域分だけ差し引く。

次に、本実施の形態の効果について説明する。

本実施の形態では、プロセス(アプリケーション)の終了処理を省略し、プロセスの終了にかかる補助記憶装置などの転送速度の遅い装置への入出力処理をなくすることができるので、システムの再起動時間を短縮することができる。

さらに、任意のプロセスを再起動後も再起動前の状態を保持したまま継続実行することができる。よって、システムの再起動に際してプロセスが提供していたサービスを強制的に中断する必要がなくなる。

次に、本発明の第 1 の実施の形態の変形例について図面を参照して詳細に説明する。

図 7 を参照すると、本発明の第 1 の実施の形態の変形例にかかる情報処理装置 100a は、再起動フラグ設定手段 134 の代わりに再起動プロセス ID テーブル 134a と再起動設定手段 134b を備えている点などで、図 2 に示した本発明の第 1 の実施の形態の情報処理装置 100 と相違する。

本発明の第 1 の実施の形態においては、OS の再起動前に、ユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させると共に、前記退避させたプロセス情報に再起動するかどうかを指定する再起動フラグ 204 を設定し、OS の再起動時に、再起動フラグ 204 が再起動しない設定になっているユーザ

プロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化し、OS 再起動後に、前記退避したプロセス情報のうち再起動フラグ 204 が再起動しない設定になっているユーザプロセスのプロセス情報を OS 内に復元した。

これに対して、第 1 の実施の形態の変形例では、再起動するユーザプロセスの ID を保持する再起動プロセス ID テーブル 134a を設け、再起動設定手段 134b が、少なくとも OS の終了前に、ユーザから入力されるプロセス ID に基づいて、再起動するユーザプロセスの ID を再起動プロセス ID テーブル 134a に設定する。プロセス情報退避手段 131 は、図 8 のフローチャートに示すように、OS の再起動前に、OS データ領域 122 から一つのプロセス表を取り出し (S110)、そのプロセス表に設定されたプロセス ID が再起動プロセス ID テーブル 134a に登録されているかどうかを調べることにより (S111)、当該プロセス表に対応するユーザプロセスが再起動するプロセスか、OS 再起動後に継続運用するユーザプロセスかを判断し (S112)、再起動せずに継続運用するユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域 124 に退避させる (S100)。すなわち、図 3 に示したようなプロセス情報退避領域 125 を作成する。但し、本変形例の場合、再起動フラグ 204 は省略してよい。そして、OS の再起動時 (S102、S103)、主記憶初期化手段 133 により、前記退避したユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化し (S104)、前記退避したプロセス情報を OS 内に復元する (S105、S106、S109)。

なお、上記の説明では、再起動プロセス ID テーブル 134a に、再起動するユーザプロセスのプロセス ID を設定したが、その反対に、再起動せずに継続運用するユーザプロセスのプロセス ID を設定するようにしてもよい。

第 1 の実施の形態では、再起動の必要なプロセスについてもプロセス情報退避領域 125 を作成していたが、本変形例では、再起動の必要のないプロセスのみプロセス情報退避領域 125 を作成するので、第 1 の実施の形態よりも、プロセス情報退避領域 125 を作成する回数を減らすことができる。

次に、本発明の第 2 の実施の形態について図面を参照して詳細に説明する。

図 9 を参照すると、本発明の第 2 の実施の形態にかかる情報処理装置 100b は、再起動フラグ設定手段 134 の代わりに、再起動プロセス ID テーブル 134a と再起動設定手段 134b を備え、プロセス情報退避手段 131 の代わりに、プロセス退避情報作成手段 135、プロセス退避情報更新手段 136 及びプロセス情報退避領域解放手段 137 を備えている点などで、図 2 に示した本発明の第 1 の実施の形態と相違する。ここで、再起動プロセス ID テーブル 134a と再起動設定手段 134b は、本発明の第 1 の実施の形態の変形例におけるものと同様のものである。つまり、再起動設定手段 134b は、ユーザから入力されるプロセス ID に基づいて、再起動するユーザプロセスの ID を再起動プロセス ID テーブル 134a に設定する。

本実施の形態では、本発明の第 1 の実施の形態におけるプロセス情報退避手段 131 を 3 つの手段に分割する。第 1 の実施の形態では、任意の時間にプロセス情報退避手段 131 を用いてプロセスの退避情報を作成していたが、本実施の形態では、プロセス生成時、プロセス切り替え時、プロセス終了時にそれぞれ、プロセス情報退避領域を作成し、更新し、領域を解放する。これら三つの処理をそれぞれプロセス退避情報作成手段 135、プロセス退避情報更新手段 136、プロセス情報退避領域解放手段 137 で実行する。プロセス切り替えとは、以下のような処理を指す。UNIX（登録商標）のようなマルチタスク OS では、プロセスは複数同時に実行されるが、ある微小時間  $\Delta t$  で CPU は一つのプロセスしか実行することができない。よってプロセスがあたかも複数同時に実行しているように見せかけるために、マルチタスク OS では、プロセスの実行を一定時間ごとに切り替えている。これをプロセス切り替えという。以下では、プロセス情報退避領域の作成、更新、領域解放について説明する。

図 10 はプロセス生成時に動作するプロセス退避情報作成手段 135 のフローチャートを示している。プロセス退避情報作成手段 135 は、OS によるプロセスの生成が完了した時点で処理を開始する。プロセスの生成完了とは、生成するプロセスのプロセス表 202 を OS データ領域 122 に用意し、当該プロセスにプロセス使用領域 123 を割り当て、その使用領域 123 に実行するプログラムをコピーし、当該プロセスが実行準備を完了した時点を目指す。まず、プロセス情報退避領域 125 を主記憶

領域 120 上に確保する(図 10 の S301)。次に、当該プロセスのプロセス表 202 を OS データ領域 122 からプロセス情報退避領域 125 にコピーする(S302)。次に、当該プロセスのブロックテーブル 203 を OS データ領域 122 からプロセス情報退避領域 125 にコピーする(S303)。最後に、単方向リスト 205 を設定し、退避領域 124 のリストに登録する(S304)。

図 11 はプロセス切り替え時に動作するプロセス退避情報更新手段 136 のフローチャートを示している。まず該当するプロセスのプロセス情報退避領域 125 を退避領域 124 から検索する(S401)。次に、ユーザから当該プロセスを再起動する旨の指示があるかどうかを再起動プロセス ID テーブル 134a を参照してチェックする(S402)。ユーザからの指示により再起動するのであれば(S403)、前記検索したプロセス情報退避領域 125 における再起動フラグ 204 を立て(S404)、プロセス退避情報更新手段 136 を終了する。再起動せず継続運用するのであれば(S403)、再起動フラグ 204 を取り消し(S405)、OS データ領域 122 上の最新のプロセス表 202 をプロセス情報退避領域 125 にコピー(S406)し、OS データ領域 122 上の最新のブロックテーブル 203 をプロセス情報退避領域 125 にコピー(S407)し、プロセス退避情報更新手段 136 を終了する。ユーザからの指示が存在しない場合は、再起動フラグに従う、再起動フラグ 204 が立っている状態であるのなら、更新処理(S406、S407)は行わず終了する。

図 12 はプロセス終了時に動作するプロセス情報退避領域解放手段 137 のフローチャートを示している。まず退避領域 124 から当該プロセスのプロセス情報退避領域 125 を検索する(S501)。次に単方向リスト 205 から当該プロセスのプロセス情報退避領域 125 を削除する(S502)。この処理(S502)は、当該プロセスのプロセス情報退避領域の前につながっている領域と後につながっている領域のリストをつなぎ合わせる処理になる。次に、当該プロセスのプロセス情報退避領域 125 を解放する(S503)。つまり、削除するプロセス情報退避領域 125 が使用している主記憶ブロックのブロック識別子 400 内の使用フラグ 402 を零(使用していない)にし、退避領域使用サイズ 201 の値から、削除するプロセス情報退避領域分だけ差し引く。これでプロセス情報退避領域解放手段 137 の処理が終了し、この後、OS によるプロセ

スの終了処理に移ることになる。

再起動後の主記憶初期化、プロセス復元などは、第 1 の実施の形態と同様である。  
本実施の形態における効果は以下の通りである。

第 1 の実施の形態では、プロセスの最新の情報を保持するためにはシステム終了時にプロセス情報退避領域 125 を作成する必要があった。例えば、システム終了 10 分前にあるプロセスについてプロセス情報退避領域 125 を作成すると、プロセス情報退避領域 125 を作成してからシステム終了までの 10 分間の間にプロセスは処理を実行し、プロセス表を書き換えたり、必要のなくなったブロックを解放し、ブロックテーブルを書き換えている可能性がある。このような場合にはプロセス復元手段 132 にてプロセス表 202、ブロックテーブル 203 を元に戻しても実行できない場合がある。よって、プロセスの最新の情報を保持するためにシステム終了時にプロセス情報退避領域 125 を作成する必要があった。しかし、本実施の形態を適用すると、常にプロセス情報を最新の状態にしておくことができ、システム終了時にプロセスの終了処理を一切行う必要がなくなる。よって、第 1 の実施の形態と比較し、より高速にシステムを終了することができる。

次に、本発明の第 3 の実施の形態について図面を参照して詳細に説明するが、その前に、第 3 の実施の形態に利用する主記憶イメージを用いた高速起動方式(以下主記憶イメージ方式と呼ぶ)について説明する。

図 1 3 を参照すると、主記憶イメージ方式を適用した情報処理装置は、少なくとも主記憶装置 101 とデータ処理装置 150 とを有し、プログラム制御により動作する。

主記憶装置 101 は、主記憶部 126 と不揮発性記憶部 140 とを含んで構成される。

不揮発性記憶部 140 は、OS 及びアプリケーションプログラムの初期化処理後の主記憶イメージを記憶する部分である。この不揮発性記憶部 140 は、不揮発なメモリなら任意の種類のメモリで良く、例えばフラッシュ RAM 等の書き換え可能な不揮発性メモリや、ROM 等の読み出し専用の不揮発性メモリが使用できる。使用する主記憶イメージの内容を変更する方法としては、不揮発性記憶部 140 の差し替え、または、不揮発性記憶部 140 が書き換え可能な場合は、ネットワークを利用したイメージファイルのダウンロードや、外部記憶媒体からのイメージファイルの読み込み

などの方法が使用可能である。

不揮発性記憶部 140 に記憶する主記憶イメージの生成方法は任意であるが、例えば、通常のシステム起動方法で起動した直後の主記憶イメージをダンプすることにより生成することができる。

主記憶部 126 は、不揮発性記憶部 140 以外の主記憶領域を構成する部分である。この主記憶部 126 は、読み書き可能なメモリなら任意の種類のメモリで良く、一般的には DRAM、SRAM 等の揮発性のメモリが使用される。但し、MRAM (Magnetoresistive RAM)、FeRAM (Ferroelectric RAM)、フラッシュ RAM 等の不揮発性のメモリを使用することも可能である。

他方、データ処理装置 150 は、マッピングテーブル変更部 190 とアドレス変換部 200 とハードウェア初期化部 180 の機能手段を備え、ハードウェア初期化部 210 はマッピングテーブル初期化部 180 を有している。またデータ処理装置 150 は、制御テーブルとしてマッピングテーブル部 160 および保護情報部 170 を有している。

マッピングテーブル部 160 は、データ処理装置 150 上で動作する OS やアプリケーションプログラム(両者を総称してプログラムと言う)が主記憶装置 101 を操作する際に指定するメモリアドレス(論理アドレス)を、主記憶装置 101 を構成する主記憶部 126 および不揮発性記憶部 140 に実際に割り当てられているメモリアドレス(物理アドレス)に変換するための情報、および保護モードを記憶する制御テーブルである。ここで、論理アドレスは論理ブロックアドレスとブロック内アドレスとで構成され、物理アドレスは物理ブロックアドレスと前記ブロック内アドレスとで構成される。

図 1 4 にマッピングテーブル部 160 のエントリの構成例を示す。マッピングテーブル部 160 は、このようなエントリの集合から構成される。図 1 4 を参照すると、1 つのエントリには、論理ブロックアドレス 161 と物理ブロックアドレス 162 と保護モード 163 の組が保持される。論理ブロックアドレス 161 は、論理アドレス空間を所定の大きさのブロックに分割したときにできる個々の論理ブロックを一意に識別するアドレスである。物理ブロックアドレス 162 は、物理アドレス空間を論理アドレス空間と同じ大きさのブロックに分割したときにできる個々の物理ブロッ

クを一意に識別するアドレスである。ブロックとしては、ページと呼ばれるアーキテクチャに固有の単位を用いることで効率良く処理することが可能となるが、主記憶イメージ方式はその単位に制限されるわけではない。保護モード 163 は、読み込み専用の保護モードと読み書き可能な保護モードの何れか一方の値をとる。この保護モード 163 は、当該情報処理装置で実行されるプログラムの本来の保護属性とは異なる擬似的なものであり、不揮発性記憶部 140 のブロックに関してはシステム起動時に読み込み専用の保護モードが一律に設定される。

保護情報部 170 は、当該情報処理装置で実行されるプログラム本来の保護属性を保持する制御テーブルである。図 15 に保護情報部 170 のエントリの構成例を示す。保護属性部 170 は、このようなエントリの集合で構成される。図 15 を参照すると、1 つのエントリには、論理ブロックアドレス 171 と保護属性 172 の組が保持される。保護属性 172 には、読み込み専用の保護属性と、読み書き可能な保護属性とがあり、何れの保護属性が設定されるかは、論理ブロックアドレス 171 で特定されるブロックの本来の保護属性によって決定される。

ハードウェア初期化部 210 は、当該情報処理装置の起動時および再起動時にシステム各部のハードウェアの初期化を行う部分であり、マッピングテーブル初期化部 180 を備えている。このマッピングテーブル初期化部 180 は、主記憶装置 101 を構成する主記憶部 126 および不揮発性記憶部 140 への物理アドレスの割り当て機能と、マッピングテーブル部 160 の初期設定機能とを有する。一般に PCI バスの制御下に置かれているメモリなどのデバイスに対しては、BIOS の機能で物理アドレスを物理的に固定させずに或る程度動的に割り当てることができるが、マッピングテーブル初期化部 180 は、このような BIOS の処理と同様な物理アドレスの割り当て機能を有している。但し、主記憶イメージ方式において物理アドレスの動的な割り当ては必須の事項でなく、物理アドレスが静的に割り当てられている計算機システムに対しても適用可能である。マッピングテーブル部 160 の初期設定では、論理アドレス空間を構成する各論理ブロックに対して、不揮発性記憶部 140 に割り当てた物理ブロックを順次に対応付け、それら全てのブロックの保護モードを、そのブロックの本来の保護属性にかかわらず読み出し専用を設定する。

アドレス変換部 200 は、当該情報処理装置上で動作するプログラムがメモリ操作の際に指定する論理アドレスを物理アドレスに変換する機能と保護モード違反による例外を検出し例外処理を起動する機能とを有する。論理アドレスから物理アドレスへの変換および保護モード違反による例外検出は、マッピングテーブル部 160 を参照して行う。マッピングテーブル変更部 190 は、アドレス変換部 200 で起動された例外処理を実行する機能を有する。例外処理では、保護情報部 170 の参照による本来の保護属性の判定、不揮発性記憶部 140 のブロックの主記憶部 126 へのコピー、マッピングテーブル部 160 の更新などの処理が行われる。

図 16 にアドレス変換部 200 およびマッピングテーブル変更部 190 の処理の一例を示す。アドレス変換部 200 は、プログラムから指定されたメモリアクセスにかかる論理アドレス中のブロックアドレスでマッピングテーブル部 160 を検索し、図 14 に示した論理ブロックアドレス 161 が一致するエントリを取得する(ステップ S701)。次に、アドレス変換部 200 は、プログラムから指定されたメモリアクセスが、書き込みアクセスか、読み出しアクセスかを判別し(ステップ S702)、読み出しアクセスであれば(ステップ S702 で NO)、ステップ S701 で取得したエントリ中の物理ブロックアドレス 162 を用いて、プログラムから指定された論理アドレスを物理アドレスに変換し(ステップ S703)、その物理アドレスで特定される主記憶装置 101 の個所をアクセスする(ステップ S704)。

プログラムから指定されたメモリアクセスが書き込みアクセスの場合(ステップ S702 で YES)、アドレス変換部 200 は、ステップ S701 で取得したエントリ中の保護モード 163 が読み込み専用の保護モードか、読み書き可能な保護モードかを判別する(ステップ S705)。読み書き可能な保護モードであれば(ステップ S705 で NO)、アドレス変換部 200 は、ステップ S701 で取得した物理ブロックアドレス 162 を用いて、プログラムから指定された論理アドレスを物理アドレスに変換し(ステップ S703)、その物理アドレスで特定される主記憶装置 101 の個所をアクセスする(ステップ S704)。

保護モード 163 が読み込み専用の保護モードである場合(ステップ S705 で YES)、アドレス変換部 200 は、当該メモリアクセスの処理を一時保留し、例外イベントを

マッピングテーブル変更部 190 に通知して例外処理を起動する(ステップ S706)。この例外イベントでは、ステップ S701 で取得したエントリ中の論理ブロックアドレス 161 と物理ブロックアドレス 162 をマッピングテーブル変更部 190 に通知する。

マッピングテーブル変更部 190 は、通知された論理ブロックアドレス 161 で保護情報部 170 を検索し、図 1 5 の論理ブロックアドレス 171 が一致するエントリの保護属性 172 を取得する(ステップ S707)。次に、マッピングテーブル変更部 190 は、保護属性 172 が読み込み専用の保護属性か、読み書き可能な保護属性かを判別する(ステップ S708)。読み込み専用の保護属性であれば(ステップ S708 で YES)、マッピングテーブル変更部 190 はアドレス変換部 170 に対してエラーを通知し、アドレス変換部 200 は保留中のメモリアクセスを拒否する等のエラー処理を行う。

他方、保護属性 172 が読み書き可能な保護属性であれば(ステップ S708 で NO)、マッピングテーブル変更部 190 は、アドレス変換部 200 から通知された物理ブロックアドレス 162 で特定される不揮発性記憶部 140 のブロックを、主記憶部 126 の空きブロックにコピーする(ステップ S709)。次にマッピングテーブル変更部 190 は、図 1 4 の論理ブロックアドレス 161 がアドレス変換部 200 から通知された論理ブロックアドレスに一致するエントリをマッピングテーブル部 160 から検索し、そのエントリの物理ブロックアドレス 162 をステップ S709 におけるコピー先の主記憶部 126 のブロックの物理ブロックアドレスに書き換え、かつ保護モード 163 を読み書き可能な保護モードに書き換える(ステップ S710)。そして、例外処理の完了をアドレス変換部 200 に通知する。アドレス変換部 200 は、マッピングテーブル変更部 190 から例外処理の完了通知を受けると、一時保留していたメモリアクセスの処理をステップ S701 から再度開始する。

図 1 7 は主記憶イメージ方式を適用した情報処理装置の概略動作を示すフローチャートである。以下、各図を参照して主記憶イメージ方式を適用した情報処理装置の動作を説明する。

情報処理装置は起動されると、図 1 7 に示されるように、まず、ハードウェア初期化部 210 によってハードウェアの初期化を行った後(ステップ S801)、マッピングテーブル初期化部 180 によって、主記憶部 126 および不揮発性記憶部 140 に物理

アドレスを割り当て、マッピングテーブル部 160 を初期設定する(ステップ S802)。マッピングテーブル部 160 の初期設定では、論理ブロックアドレスと物理ブロックアドレスの対応関係をマッピングテーブル部 160 に書き込み、且つ、不揮発性記憶部 140 の全てのブロックは読み込み専用の保護モードでマッピングテーブル部 160 に登録する。

次にデータ処理装置 150 は、不揮発性記憶部 140 に保存されている主記憶イメージからシステムを起動する(ステップ S803)。この主記憶イメージは、OS やアプリケーションの起動直後と同じ状態であるため、システムは立ち上げ直後の環境に復帰することになる。このとき、主記憶イメージから起動してから、システム運用(ステップ S804)が可能になるまでの運用中断状態は存在しない。

システム運用(ステップ S804)の段階において、プログラムから不揮発性記憶部 140 のとあるブロックに対して初めての書き込みアクセスが発生したときには、そこは読み込み専用の保護モードでマッピングされているので、アドレス変換部 200 が例外イベントを通知し、当該書き込みアクセスを一時保留する(図 16 の S706)。例外イベントを受け取ったマッピングテーブル変更部 190 は、保護情報部 170 を参照してその書き込み発生論理アドレスのブロックの保護属性を判断し(図 16 のステップ S708)、読み書き可能属性であれば、そのブロックの大きさの空きブロックを主記憶部 126 から割り当て、その不揮発性記憶部 140 の当該ブロックの内容をコピーする(図 16 のステップ S709)。更に、マッピングテーブル変更部 190 は、割り当てられた主記憶ブロックを書き込みが発生した論理ブロックアドレスに読み書き可の保護モードでマッピングするようマッピングテーブル部 160 に設定して、一時保留された書き込みアクセスの処理を再開させる(図 16 のステップ S710)。この結果、一時保留された書き込みアクセスおよびその後の当該ブロックへの書き込み操作は、割り当てられた主記憶ブロックに対して実行される。このとき、ブロックの単位を小さくすることでコピー時間が短くなり、システム運用が中断される時間を短くすることができる。

この状態で、再起動が行われると(図 17 のステップ S805)、再びハードウェアの初期化処理(ステップ S801)から実行され、起動時と同じようにマッピングテー

ブル初期化部 180 の動作によってマッピングテーブル部 160 の内容が初期化され、不揮発性記憶部 140 がマッピングし直される(ステップ S802)。このため、システムの運用中に主記憶部 126 に対して行われた書き込みの内容は一切反映されていない状態に戻り、これがつまり、立ち上げ直後の環境に復帰することと同じになる。

このように主記憶イメージ方式を適用した情報処理装置にあっては、不揮発性記憶部 140 に、OS 及びアプリケーションプログラムの初期化処理後の主記憶イメージが記憶されており、データ処理装置 150 は、不揮発性記憶部 140 に記憶された主記憶イメージからシステムを起動するため、システムの高速起動が可能になる。また、システム運用中に OS またはアプリケーションプログラムから不揮発性記憶部 140 への書き込みアクセスが発生すると、データ処理装置 150 は、書き込みアクセスが発生したアドレスを含む所定幅のアドレス範囲内のデータを不揮発性記憶部 140 から主記憶部 126 に確保した代替領域にコピーし且つ以後の前記所定幅のアドレス範囲へのアクセスを前記代替領域へのアクセスに変換する設定を行う。このため、前記書き込みアクセスおよびそれ以降の前記所定幅のアクセス範囲へのアクセスは、前記代替領域へのアクセスに変換され、代替領域上のコピーが更新される。このように、OS やアプリケーションプログラムによって書き換える必要のあるデータをシステムの起動時に不揮発性記憶部 140 から読み書き可能な主記憶部 126 へ連続してコピーしておくのではなく、書き込みアクセスが発生した都度実施することにより、システム起動後に直ちにシステムの運用を開始することができる。

以上が主記憶イメージを用いた高速再起動方式の説明である。続いて、この高速起動方式を応用した本発明の第 3 の実施の形態について図面を参照して説明する。

図 18 を参照すると、本発明の第 3 の実施の形態にかかる情報処理装置 100c は、主記憶領域 120 が主記憶部 126 と不揮発性記憶領域 140 で構成され、データ処理装置 150 が追加されている点で、図 2 に示した第 1 の実施の形態の情報処理装置 100 と構成が相違する。

不揮発性記憶部 140 には、図 13 の不揮発性記憶部 140 の場合と異なり、OS のみの起動イメージだけが記憶されている。つまり、OS を通常起動した直後の主記憶上の内容を記憶している。不揮発性記憶部 140 に記憶される OS の主記憶イメー

ジ(起動イメージ)には、OS プログラム領域と OS データ領域とが含まれる。主記憶部 126 内にも OS データ領域 122 が図示されているが、この OS データ領域 122 は不揮発性記憶部 140 内の OS データ領域 122 が運用中に書き込みのためにアクセスされたことによって、図 1 6 のステップ S709 でコピーされて生成された領域であり、前述した「代替領域」に相当する。

データ処理装置 150 は、図 1 3 で説明したような構成を備え、プログラムを実行する際のアドレス変換などを担う。

本実施の形態では、第 1 の実施の形態の動作を説明した図 4 A 及び図 4 B の OS 起動部分(S103)に、OS の主記憶イメージを用いた高速起動方式を用いる。また、本実施の形態では、主記憶初期化手段 133 の処理内容が第 1 の実施の形態と異なる。すなわち、第 1 の実施の形態と同様、OS に対し既に使用されているブロックを認識させる必要があるが、主記憶イメージはすでに主記憶領域の初期化を終了したものとみなすことができる。よって、第 1 の実施の形態における主記憶領域初期化手段 133 とは異なり、ブロック識別子 400 のための領域を主記憶上に確保する(S202)が必要がなく、すでに存在するブロック識別子に関して使用されているか否かを確認するのみでよい。また、第 1 の実施の形態と同様に、退避領域使用サイズ 201 とプロセスが使用しているブロックのリストは事前に知っておく必要がある。

図 1 9 に本実施の形態における主記憶初期化手段 133 のフローチャートを示す。第 1 の実施の形態との違いは、ブロック識別子のための主記憶領域を用意し 0 で初期化するか(S204)、すでに存在するブロック識別子を検索するか(S904)である。

本実施の形態によれば、OS の起動に OS 起動イメージを利用するため、OS 起動時の初期化処理および転送速度の遅い補助記憶装置からの入出力をなくすることができるので、第 1 の実施の形態よりもさらに高速な再起動を実現することができる。勿論、常に再起動されるのは OS だけであり、再起動前に実行中のアプリケーションは、再起動フラグの値に応じて、再起動されるか、継続運用されるかが決定される。

本実施の形態は、第 1 の実施の形態に主記憶イメージによる高速起動方式を適用したものであるが、第 1 の実施の形態の変形例(図 7)または第 2 の実施の形態に主

記憶イメージによる高速起動方式を適用し、OS の再起動の高速化を図るようにすることもできる。

次に、本発明の第 4 の実施の形態について図面を参照して詳細に説明する。

図 20 を参照すると、本発明の第 4 の実施の形態にかかる情報処理装置 100d は、図 2 の第 1 の実施の形態にかかる情報処理装置 100 と比較して、再起動フラグ設定手段 134 に代えて、再起動設定受付手段 220、再起動プロセス ID テーブル 221 及び再起動フラグ設定手段 222 を備えている点で相違する。

再起動設定受付手段 220 は、ユーザから再起動するプロセスのプロセス ID をキーボード等を通じて受け付け、再起動プロセス ID テーブル 221 に保存する手段である。再起動フラグ設定手段 222 は、第 1 の実施の形態における再起動フラグ設定手段 134 と同様に、プロセス情報退避手段 131 によって退避領域 124 に生成されたプロセス情報退避領域 125 に設けられた再起動フラグ 204 の設定を、システム終了までの任意の時間に行う手段であるが、再起動プロセス ID テーブル 221 にプロセス ID が保存されたユーザ指定の再起動プロセスだけでなく、その再起動プロセスと同じアプリケーションに属するすべてのプロセスの再起動フラグ 204 をサーチし、これらのプロセスの再起動フラグ 204 も同様に設定する機能を有する。

一般に、複数のプロセスによってアプリケーションが構成される場合、それら複数のプロセスには、同じプロセスグループ ID が付与され、プロセス表 202 の 1 エントリに保存される。従って、或るプロセスと同じアプリケーションに属する他のプロセスがどれであるかは、プロセス表 202 に保存されているプロセスグループ ID を調べることで可能である。

図 21 に再起動フラグ設定手段 222 の処理のフローチャートを示す。まず再起動フラグ設定手段 222 は、再起動プロセス ID テーブル 221 にプロセス ID が保存されている 1 つのプロセスに注目し(ステップ S1001)、そのプロセスに対応するプロセス情報退避領域 125 の再起動フラグ 204 をアクセスし、立っていないければ立てる(ステップ S1002)。次に、注目中のプロセスのプロセス表 202 からプロセスグループ ID を取得し(ステップ S1003)、他のプロセス情報退避領域 125 のプロセス表 202 のプロセスグループ ID を調べて、同じプロセスグループ ID を持つ他のプロセスす

べてを検索する(ステップ S1004)。そして、検索したすべてのプロセスのプロセス情報退避領域 125 の再起動フラグ 204 をアクセスし、立っていないならば立てる(ステップ S1005)。次に、再起動プロセス ID テーブル 221 にプロセス ID が保存されている他の 1 つのプロセスに注目を移し(ステップ S1006)、ステップ S1002 に戻って上述した処理と同様の処理を繰り返す。再起動プロセス ID テーブル 221 に保存されたすべてのプロセス ID に注目した処理を終えた時点で(ステップ S1007 で YES)、処理を終える。

このように本実施の形態によれば、ユーザは、アプリケーションを構成する複数のプロセスのうち少なくとも 1 つのプロセスを再起動プロセスとして指定しておけば、残りのプロセスを再起動プロセスとして明示的に指定しなくても、自動的に再起動プロセスとして処理される。通常アプリケーションは複数のプロセスから構成されることが多く、1 つのアプリケーションを構成する複数のプロセスをこのように一括管理して再起動の設定を施すことで、個々のプロセスごとに再起動の設定を施すという煩雑な作業からユーザを解放することができる。また、再起動後にアプリケーションを構成するプロセスが足りない(アプリケーションの中の 1 つのプロセスのみ再起動してしまった。)などというミスを防ぐことができる。

本実施の形態は、第 1 の実施の形態を前提としたが、同様の考えを第 2 の実施の形態あるいは第 3 の実施の形態に対して適用することが可能である。

次に本発明の第 1 の実施例を、図面を参照して説明する。かかる実施例は本発明における第 1 の実施の形態に対応するものである。

本実施例は具体的な OS の例として、Linux を使用する。

Linux において、プロセスを管理するためにカーネルが持っている情報としては、プロセスディスクリプタと、このプロセスディスクリプタに付随する他の構造体がある。プロセスディスクリプタは、プロセスを実行するために必要なほとんどの情報を保持している。他の構造体とは、例えばファイルをオープンしていれば、ファイル構造体、ソケットをオープンし、他の計算機と通信を行えば、ソケット構造体がプロセスディスクリプタに付随する。プロセスディスクリプタには、これら付随する構造体へのポインタが格納されている。これらプロセスディスクリプタと付随

する構造体が、図 3 のプロセス表 202 にあたる。

また、Linux では、主記憶領域 120 をページという単位で管理している。プロセスの使用している主記憶領域 120 は複数のページから構成され、複数のページをまとめて管理するためにプロセス 1 つにつき 1 つのページテーブルが OS により用意される。このページテーブルが、ブロックテーブル 203 にあたる。

プロセスを途中で中断し、再起動後にプロセスを継続実行するには、上記のプロセスディスクリプタ、ページテーブル、プロセスの使用メモリ領域のほかに、当該プロセスが実行中であれば、CPU 内のレジスタに格納されているデータが必要となる。マルチタスク OS ではプロセスを複数同時に実行するためにプロセス切り替えを行っている。

プロセス切り替えによって、実行を一時中断されたプロセスの CPU 内のデータは、タスク状態セグメントとカーネルモードスタックに分割して格納される。タスク状態セグメントは、プロセスディスクリプタの中にあるエントリの一つであり、カーネルモードスタックは、プロセスが使用しているメモリ領域の一部である。プロセスを再開するときは、タスク状態セグメントと、カーネルモードスタックから中断されたときの CPU レジスタの状態を復元し実行を再開する。

プロセス情報退避手段 131 では、上記に記載したプロセスに関する情報(プロセスディスクリプタ、ページテーブルなど)をプロセス情報退避領域 125 に退避させることである。図 2 2 はプロセス情報退避手段 131 の Linux での処理を表している。

生成された全てのプロセスのためのプロセスディスクリプタは、リスト構造になっており、プロセスディスクリプタの先頭アドレスは、init\_task という変数に収められている。まず、init\_task からプロセスディスクリプタのアドレスを取得する(ステップ S1301)。次にプロセス情報退避領域 125 を主記憶上に確保し(ステップ S1302)、プロセスディスクリプタを作成したプロセス情報退避領域 125 にコピーする(ステップ S1303)。同様に、プロセスディスクリプタに付随する他の構造体、ページテーブルもプロセス情報退避領域 125 にコピーする(ステップ S1304、S1305)。プロセス情報退避領域 125 を単方向リスト 205 に登録し(ステップ S1306)、退避領域使用サイズ 201 に使用した主記憶サイズを足す(ステップ S1307)。

以上でプロセス情報退避手段 131 の一度の処理が終了する。この処理を生成されたプロセス全てについて実行する。プロセス全てについてプロセス情報退避領域 125 を作成したのち Linux を通常終了させる。

システム起動時は主記憶初期化とプロセス復元の二つの処理を行う。

Linux では、主記憶をページという単位で管理している。ページにはページディスクリプタとよばれる当該ページに関する情報を格納している領域があり、起動時に Linux では、Linux 自身とハードウェアによって使用されている領域を除き、すべての主記憶領域 120 をページ単位に分割し、その各々にページディスクリプタを割り当てることで、利用できる主記憶領域 120 の初期化を行う。このような処理は、主記憶領域 120 をある単位（ページやセグメントなど）で管理している OS に共通する処理である。主記憶初期化処理の時点で、再起動を行わないプロセスについては、当該プロセスが使用している主記憶空間を OS に認識させる必要がある。本実施例におけるページは、第 1 の実施の形態におけるブロックに相当し、ページディスクリプタはブロック識別子 400 に相当する。

Linux は、上述したとおり主記憶領域 120 をページという単位で管理しており、主記憶上のページの一つ一つにページディスクリプタが存在する。Linux では、このページディスクリプタの初期化作業をもって、主記憶領域 120 の初期化としている。Linux の実装では、ページディスクリプタは、OS 予約領域と呼ばれる OS をメモリ上に展開するために用意された領域の直後に配置され、固定長である。よって、再起動後もページディスクリプタを配置している主記憶上の位置、大きさは変わらない。また、主記憶領域 120 のページが使用されているかどうかは、ページディスクリプタ内の count 変数によって管理されている。count 変数が 1 以上の時に当該ページは使用されていることになる。共有メモリを使用する複数のプロセスは、1 つのページを複数のプロセスで使用するようになる。このような場合には、count 変数は 1 以上の値をとることになる。

図 2 3 に本実施例における主記憶初期化手段 133 の処理を示す。まず、退避領域 124 の使用している主記憶領域 120 を退避領域使用サイズ 201 から確認しておく（ステップ S1401）。次に退避領域 124 内にある全てのプロセス情報退避領域 125 内

のページテーブルを検索し、どのページがどのプロセスによって利用されてるかどうかを調べておく(ステップ S1402)。調べた結果は、ページディスクリプタ番号と使用しているプロセス数のテーブルにして保持しておく。

主記憶領域 120 の最初のアドレス(厳密には、OS の予約領域の直後のアドレス)から、ページディスクリプタを設置していく。指定されたアドレスにつき一つのディスクリプタ領域を用意し 0 で初期化する(ステップ S1404)。次に、指定されたアドレスが退避領域 124 に格納されているページテーブルで使用されているかを確認する(ステップ S1405)。確認には上記にて調べたページディスクリプタ番号と使用しているプロセス数のテーブルを利用する。使用していれば(ステップ S1406 の YES)、ページディスクリプタ内の count 変数に使用しているプロセス数を代入する(ステップ S1407)。使用していない場合(ステップ S1406 の NO)もしくはステップ S1407 の処理を終了すると、次のページディスクリプタの初期化に移る。利用できるすべての主記憶領域 120 を初期化すると(ステップ S1403 の YES)、主記憶初期化手段 133 の処理が完了する。

プロセス復元手段 132 は、プロセスディスクリプタおよび付随する他の構造体ならびにページテーブルを復元のために用意した OS データ領域 122 にコピーすることであり、実施の形態 1 に記載したプロセス復元処理と同様である。

次に、第 2 の実施の形態について実施例を以下に示す。

図 2 4 はプロセス生成時のプロセス退避情報作成手段 135 の処理内容を示している。

ユーザによりプログラムの実行要求が OS に指示されると、OS はプロセスを生成する。ここでは、ある一つのプロセスの名称としてプロセス A と呼ぶ。プロセス A が生成され、プロセスディスクリプタがプロセス実行キューに登録されることで初めてプロセス A が実行される。プロセス退避情報生成手段 135 は、プロセス実行キューに当該プロセスが登録される前に実行する。プロセス A が生成された時点で、プロセスディスクリプタが格納されているアドレスは、すでに OS が保持している(OS 内の一時変数に生成したプロセス A のプロセスディスクリプタが格納されている)。よって、プロセスディスクリプタのアドレスを取得する処理は行わない。ま

ず最初にプロセス A の退避情報を格納する領域をメモリ上に確保する(ステップ S1501)。次に、プロセス A のプロセスディスクリプタを確保したプロセス情報退避領域 125 にコピーする(ステップ S1502)。さらにプロセスディスクリプタに付随する他の構造体(ファイル構造体、ソケット構造体など)を同退避領域 125 にコピーする(ステップ S1503)。次に、プロセス A のページテーブルを同退避領域 125 にコピーする(ステップ S1504)。ページテーブルが置かれているアドレスは、プロセスディスクリプタ中のメモリディスクリプタに格納されているので、それを参照する。最後に同退避領域 125 を退避領域リストに登録し、退避領域サイズに使用した主記憶サイズを足す(ステップ S1505、S1506)。

図 25 にプロセス退避情報更新手段 136 の処理内容を示す。更新処理は、プロセス切り替えに関する処理がすべて終了し、次に実行されるプロセスのレジスタ情報がタスク状態セグメントから復元される前に行う。プロセス A のプロセス情報退避領域 125 の更新処理を行う前に、まずユーザから当該プロセスを再起動するかどうかの命令を受け取っているかを確認する。この命令はシステムコールとして実装されてもよいし、他の方法を使っても良い。ユーザから当該プロセスを再起動するとの指示があると(ステップ S1601 の YES)、更新処理としては、プロセス A のプロセス情報退避領域 125 にある再起動フラグ 203 を立てて、処理を終了する(ステップ S1607)。ユーザから再起動が指示されていなければ(ステップ S1601 の NO)、以下のように処理を行う。

まず、プロセス A のスタックポインタよりプロセスディスクリプタのアドレスを取得する(ステップ S1602)。次に、退避領域 124 からプロセス A のプロセス情報退避領域 125 のアドレスを取得する。図 3 における退避領域 124 のデータ構造を参照するとわかるように、退避領域 124 は、各プロセスのプロセス情報退避領域 125 をリストで管理している。よって、一番先頭(リストの最初のプロセス情報退避領域 125)のアドレスを取得し、つぎにリストを探索していくことで目的のプロセスのプロセス情報退避領域 125 のアドレスが取得できる。現在、アクセスしている退避領域 124 がどのプロセスのプロセス情報退避領域 125 かを確認するには、プロセス生成時点でコピーされているプロセスディスクリプタ内のプロセス ID を見れば

よい。プロセス A の退避領域 124 のアドレスを取得できれば、次にプロセス A のプロセスディスクリプタと付随する構造体をプロセス A のプロセス情報退避領域 125 にコピーする(ステップ S1604、S1605)。最後に、ページテーブルをコピーする(ステップ S1606)。

プロセス終了時のプロセス情報退避領域解放手段 137 の処理内容は、第 2 の実施の形態と同様である。まず、退避領域 124 のリストからプロセス A のプロセス情報退避領域 125 のアドレスを取得する。アドレスが取得できたら、プロセス情報退避領域 125 をリストから削除し、領域を解放する。その後プロセスの終了処理を行う。

以上で OS の持つプロセス情報の退避処理がすべて完了する。このように計算機システムの運用中に退避情報を作成しておくことで、再起動時には終了処理を行わなくてすむ。よって、終了処理に伴う補助記憶装置へのアクセスの削減に伴って再起動時間の短縮が実現できる。

#### 産業上の利用可能性

本発明の第 1 の効果は、システムの終了時間を削減できることにある。その理由は次の通りである。システムの終了処理とは、OS の終了処理とアプリケーションの終了処理から構成される。OS は通常終了することになるが、アプリケーションの終了処理に関しては、当該アプリケーションを構成するプロセスのプロセス表などのプロセス情報を主記憶上の退避領域に退避させているだけである。よって、アプリケーションの終了処理に伴う転送速度の遅い補助記憶装置に対するアクセスを軽減することになるので、システムの終了時間を短縮することができる。

本発明の第 2 の効果は、再起動後も再起動前に運用していたアプリケーションを継続実行できることにある。その理由は、再起動前に OS の使用している主記憶上よりプロセス表などのプロセス情報を退避し、再起動後には OS の使用している主記憶上にプロセス表などのプロセス情報を復元するため、プロセス表を退避、復元したプロセスは、実行を中断、再開したに過ぎず、あたかも継続運用されているように見えるためである。よって、障害の発生していないアプリケーションは、サービスを継続提供することができる。

よって、本発明によれば、再起動時にユーザへのサービスを中断することがないので、長時間連続運用が求められる大規模アプリケーションサーバーといった用途に適用できる。また、通常のパーソナル情報処理装置上の OS、たとえば Windows などにおけるレジストリ書き換えによる再起動の際にも、ユーザの作業状況はそのままに、OS のみ再起動することにより書き換えたレジストリ情報を反映させるような用途として適用可能である。

## 請 求 の 範 囲

1. OS の再起動前に、ユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させ、OS の再起動時に、前記ユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化し、OS 再起動後に、前記退避したプロセス情報を OS 内に復元することを特徴とする高速再起動方法。

2. OS の再起動前に、ユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させると共に、前記退避させたプロセス情報に再起動するかどうかを指定する再起動フラグを設定し、OS の再起動時に、前記再起動フラグが再起動しない設定になっているユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化し、OS 再起動後に、前記退避したプロセス情報のうち前記再起動フラグが再起動しない設定になっているユーザプロセスのプロセス情報を OS 内に復元することを特徴とする高速再起動方法。

3. OS の再起動前に、OS 再起動後に継続運用するユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させ、OS の再起動時に、前記ユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化し、OS 再起動後に、前記退避したプロセス情報を OS 内に復元することを特徴とする高速再起動方法。

4. OS の再起動前に、OS の再起動後に継続運用するプロセスまたは継続運用しないプロセスの識別子を保存してあるプロセス ID テーブルを参照して、OS 再起動後に継続運用するユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させ、OS の再起動時に、前記ユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化し、OS 再起動後に、前記退避したプロセス情報を OS 内に復元することを特徴とする高速再起動方法。

5. ユーザプロセスの生成時、前記生成したユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させ、前記ユーザプロセスの切替え時、前記退避させたプロセス情報に再起動するかどうかを指定する再起動フラグを設定するとともに、再起動しない場合には退避領域に退避されているプロセス情報を最新の状態に更新し、前記ユーザプロセスの終了時、前記退避させたプロセス情報を無効にし、OS の再起動時に、前記再起動フラグが再起動しない設定になっているユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化し、OS 再起動後に、前記退避したプロセス情報のうち前記再起動フラグが再起動しない設定になっているユーザプロセスのプロセス情報を OS 内に復元することを特徴とする高速再起動方法。

6. 或るユーザプロセスに関するプロセス情報に再起動するかどうかを指定する再起動フラグを設定する際、そのユーザプロセスと同じユーザアプリケーションプログラムに属する他の全てのユーザプロセスを検索し、該検索した全てのユーザプロセスに関するプロセス情報中の再起動フラグも同じ値に設定することを特徴とする請求の範囲第 2 項に記載の高速再起動方法。

7. 或るユーザプロセスに関するプロセス情報に再起動するかどうかを指定する再起動フラグを設定する際、そのユーザプロセスと同じユーザアプリケーションプログラムに属する他の全てのユーザプロセスを検索し、該検索した全てのユーザプロセスに関するプロセス情報中の再起動フラグも同じ値に設定することを特徴とする請求の範囲第 5 項に記載の高速再起動方法。

8. 主記憶装置の一部を構成する不揮発性記憶部に記憶された OS の主記憶イメージから OS の起動を行うことを特徴とする請求の範囲第 1 項に記載の高速再起動方法。

9. システム運用中に OS から前記不揮発性記憶部への書き込みアクセスが発生する毎に、書き込みアクセスが発生したアドレスを含む所定幅のアドレス範囲内のデータを前記不揮発性記憶部から主記憶装置の一部を構成する読み出し書き込み可能な主記憶部に確保した代替領域にコピーし且つ以後の前記所定幅のアドレス範囲へのアクセスを前記代替領域へのアクセスに変換することを特徴とする請求の範囲第 8 項に記載の高速再起動方法。

10. OS の再起動前に、ユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス情報退避手段と、OS の再起動時に、前記ユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段と、OS 再起動後に、前記退避したプロセス情報を OS 内に復元するプロセス復元手段とを備えることを特徴とする情報処理装置。

11. OS の再起動前に、ユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス情報退避手段と、前記退避させたプロセス情報に再起動するかどうかを指定する再起動フラグを設定する再起動フラグ設定手段と、OS の再起動時に、前記再起動フラグが再起動しない設定になっているユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段と、OS 再起動後に、前記退避したプロセス情報のうち前記再起動フラグが再起動しない設定になっているユーザプロセスのプロセス情報を OS 内に復元するプロセス復元手段とを備えることを特徴とする情報処理装置。

12. OS の再起動前に、OS 再起動後に継続運用するユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス情報退避手段と、OS の再起動時に、前記ユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段と、OS 再起動後

に、前記退避したプロセス情報を OS 内に復元するプロセス復元手段とを備えることを特徴とする情報処理装置。

13. 前記プロセス情報退避手段は、OS の再起動前に、OS の再起動後に継続運用するプロセスまたは継続運用しないプロセスの識別子を保存するプロセス ID テーブルを参照して、OS 再起動後に継続運用するユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるものであることを特徴とする請求の範囲第12項に記載の情報処理装置。

14. ユーザプロセスの生成時、前記生成したユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス退避領域作成手段と、前記ユーザプロセスの切替え時、前記退避させたプロセス情報に再起動するかどうかを指定する再起動フラグを設定するとともに、再起動しない場合には退避領域に退避されているプロセス情報を最新の状態に更新するプロセス退避情報更新手段と、前記ユーザプロセスの終了時、前記退避させたプロセス情報を無効にするプロセス情報退避領域解放手段と、OS の再起動時に、前記再起動フラグが再起動しない設定になっているユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段と、OS 再起動後に、前記退避したプロセス情報のうち前記再起動フラグが再起動しない設定になっているユーザプロセスのプロセス情報を OS 内に復元するプロセス復元手段とを備えることを特徴とする情報処理装置。

15. 或るユーザプロセスに関するプロセス情報に再起動するかどうかを指定する再起動フラグを設定する際、そのユーザプロセスと同じユーザアプリケーションプログラムに属する他の全てのユーザプロセスを検索し、該検索した全てのユーザプロセスに関するプロセス情報中の再起動フラグも同じ値に設定する手段を備えることを特徴とする請求の範囲第11項に記載の情報処理装置。

16. 或るユーザプロセスに関するプロセス情報に再起動するかどうかを指定する再起動フラグを設定する際、そのユーザプロセスと同じユーザアプリケーションプログラムに属する他の全てのユーザプロセスを検索し、該検索した全てのユーザプロセスに関するプロセス情報中の再起動フラグも同じ値に設定する手段を備えることを特徴とする請求の範囲第14項に記載の情報処理装置。

17. 主記憶装置の一部を構成する不揮発性記憶部に記憶されたOSの主記憶イメージからOSの起動を行う手段を備えることを特徴とする請求の範囲第10項に記載の情報処理装置。

18. システム運用中にOSから前記不揮発性記憶部への書き込みアクセスが発生する毎に、書き込みアクセスが発生したアドレスを含む所定幅のアドレス範囲内のデータを前記不揮発性記憶部から主記憶装置の一部を構成する読み出し書き込み可能な主記憶部に確保した代替領域にコピーし且つ以後の前記所定幅のアドレス範囲へのアクセスを前記代替領域へのアクセスに変換する手段を備えることを特徴とする請求の範囲第17項に記載の情報処理装置。

19. コンピュータを、OSの再起動前に、ユーザプロセスに関するOS内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス情報退避手段、OSの再起動時に、前記ユーザプロセスの使用していた主記憶領域は初期化せずにOSの使用していた主記憶領域は初期化する主記憶初期化手段、OS再起動後に、前記退避したプロセス情報をOS内に復元するプロセス復元手段、として機能させることを特徴とするプログラム。

20. コンピュータを、OSの再起動前に、ユーザプロセスに関するOS内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス情報退避手段、前記退避させたプロセス情報に再起動するかどうかを指定する再起動フラグを設定する再起動フラグ設定手段、OSの再起動時に、前記再起動フラグが再起動しない設定に

なっているユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段、OS 再起動後に、前記退避したプロセス情報のうち前記再起動フラグが再起動しない設定になっているユーザプロセスのプロセス情報を OS 内に復元するプロセス復元手段、として機能させることを特徴とするプログラム。

2 1. コンピュータを、OS の再起動前に、OS 再起動後に継続運用するユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス情報退避手段、OS の再起動時に、前記ユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段、OS 再起動後に、前記退避したプロセス情報を OS 内に復元するプロセス復元手段、として機能させることを特徴とするプログラム。

2 2. 前記プロセス情報退避手段は、OS の再起動前に、OS の再起動後に継続運用するプロセスまたは継続運用しないプロセスの識別子を保存するプロセス ID テーブルを参照して、OS 再起動後に継続運用するユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるものであることを特徴とする請求の範囲第 2 1 項に記載のプログラム。

2 3. コンピュータを、ユーザプロセスの生成時、前記生成したユーザプロセスに関する OS 内のプロセス情報を主記憶装置上の退避領域に退避させるプロセス退避領域作成手段、前記ユーザプロセスの切替え時、前記退避させたプロセス情報に再起動するかどうかを指定する再起動フラグを設定するとともに、再起動しない場合には退避領域に退避されているプロセス情報を最新の状態に更新するプロセス退避情報更新手段、前記ユーザプロセスの終了時、前記退避させたプロセス情報を無効にするプロセス情報退避領域解放手段、OS の再起動時に、前記再起動フラグが再起動しない設定になっているユーザプロセスの使用していた主記憶領域は初期化せずに OS の使用していた主記憶領域は初期化する主記憶初期化手段、OS 再起

動後に、前記退避したプロセス情報のうち前記再起動フラグが再起動しない設定になっているユーザプロセスのプロセス情報を OS 内に復元するプロセス復元手段、として機能させることを特徴とするプログラム。

24. コンピュータを、さらに、或るユーザプロセスに関するプロセス情報に再起動するかどうかを指定する再起動フラグを設定する際、そのユーザプロセスと同じユーザアプリケーションプログラムに属する他の全てのユーザプロセスを検索し、該検索した全てのユーザプロセスに関するプロセス情報中の再起動フラグも同じ値に設定する手段、として機能させることを特徴とする請求の範囲第20項に記載のプログラム。

25. コンピュータを、さらに、或るユーザプロセスに関するプロセス情報に再起動するかどうかを指定する再起動フラグを設定する際、そのユーザプロセスと同じユーザアプリケーションプログラムに属する他の全てのユーザプロセスを検索し、該検索した全てのユーザプロセスに関するプロセス情報中の再起動フラグも同じ値に設定する手段、として機能させることを特徴とする請求の範囲第23項に記載のプログラム。

26. コンピュータを、さらに、主記憶装置の一部を構成する不揮発性記憶部に記憶された OS の主記憶イメージから OS の起動を行う手段、として機能させることを特徴とする請求の範囲第19項に記載のプログラム。

27. コンピュータを、さらに、システム運用中に OS から前記不揮発性記憶部への書き込みアクセスが発生する毎に、書き込みアクセスが発生したアドレスを含む所定幅のアドレス範囲内のデータを前記不揮発性記憶部から主記憶装置の一部を構成する読み出し書き込み可能な主記憶部に確保した代替領域にコピーし且つ以後の前記所定幅のアドレス範囲へのアクセスを前記代替領域へのアクセスに変換する手段、として機能させることを特徴とする請求の範囲第26項に記載のプロ

グラム。

28. メインメモリ上に、OSをロードするためのメモリ領域である第1のOSメモリ領域と、プロセスをロードするためのメモリ領域であるプロセスメモリ領域とを割り当てて、OS及びプロセスを該当する領域にロードしたコンピュータで、OSを再起動する方法において、

OSがプロセスを管理するための情報であるプロセス情報を、前記第1のOSメモリ領域から取得し、予め定められた記憶装置に設けた退避領域に格納する段階1と、

前記プロセスメモリ領域を保持したまま、前記第1のOSメモリ領域を初期化する段階2と、

前記メインメモリ上に第2のOSメモリ領域を割り当てて、前記OSをロードする段階3と、

OSメモリ領域中のプロセス情報を、前記段階1にて格納されたプロセス情報に応じて更新する段階4と  
を含むことを特徴とするOS再起動方法。

29. 請求の範囲第28項に記載のOS再起動方法において、更に、  
プロセスメモリ領域にロードされたプロセスから保持すべきプロセスを選択する段階と、

選択されなかったプロセスに割り当てられたプロセスメモリ領域を初期化する段階と  
を含むことを特徴とするOS再起動方法。

30. 請求の範囲第28項に記載のOS再起動方法において、前記退避領域をメインメモリ上に設けることを特徴とするOS再起動方法。

31. 請求の範囲第28項に記載のOS再起動方法において、各プロセスを再起

動するか否かを示す情報を、当該プロセスのプロセス情報と共に、前記退避領域に記憶することを特徴とするOS再起動方法。

32. 請求の範囲第28項に記載のOS再起動方法において、各プロセスを再起動するか否かを示す情報を、前記退避領域を設けた記憶装置とは別の記憶装置に記憶することを特徴とするOS再起動方法。

33. 請求の範囲第28項に記載のOS再起動方法において、プロセスの生成、切り替え及び終了に応じて、前記退避領域を生成、更新及び解放する処理を、前記退避領域を設けた記憶装置に対して実行することを特徴とするOS再起動方法。

34. 請求の範囲第28項に記載のOS再起動方法において、  
前記OSをメインメモリにロードしたときのイメージを格納した不揮発性記憶装置を予め用意する方法であって、  
前記段階3は、前記不揮発性記憶装置に格納したイメージを参照して前記メインメモリに前記OSをロードすることを特徴とするOS再起動方法。

35. 請求の範囲第28項に記載のOS再起動方法において、更に、  
一のアプリケーションプログラムに対して関連付けられた複数のプロセスを含むプロセスを、プロセスメモリ領域にロードする段階と、  
プロセスメモリ領域にロードされたプロセスの中から、保持すべきプロセスを選択する段階と、  
選択されたプロセス、及び、当該プロセスと同じアプリケーションプログラムに関連付けられた他のプロセス以外のプロセスに割り当てられたプロセスメモリ領域を、初期化する段階とを含むことを特徴とするOS再起動方法。

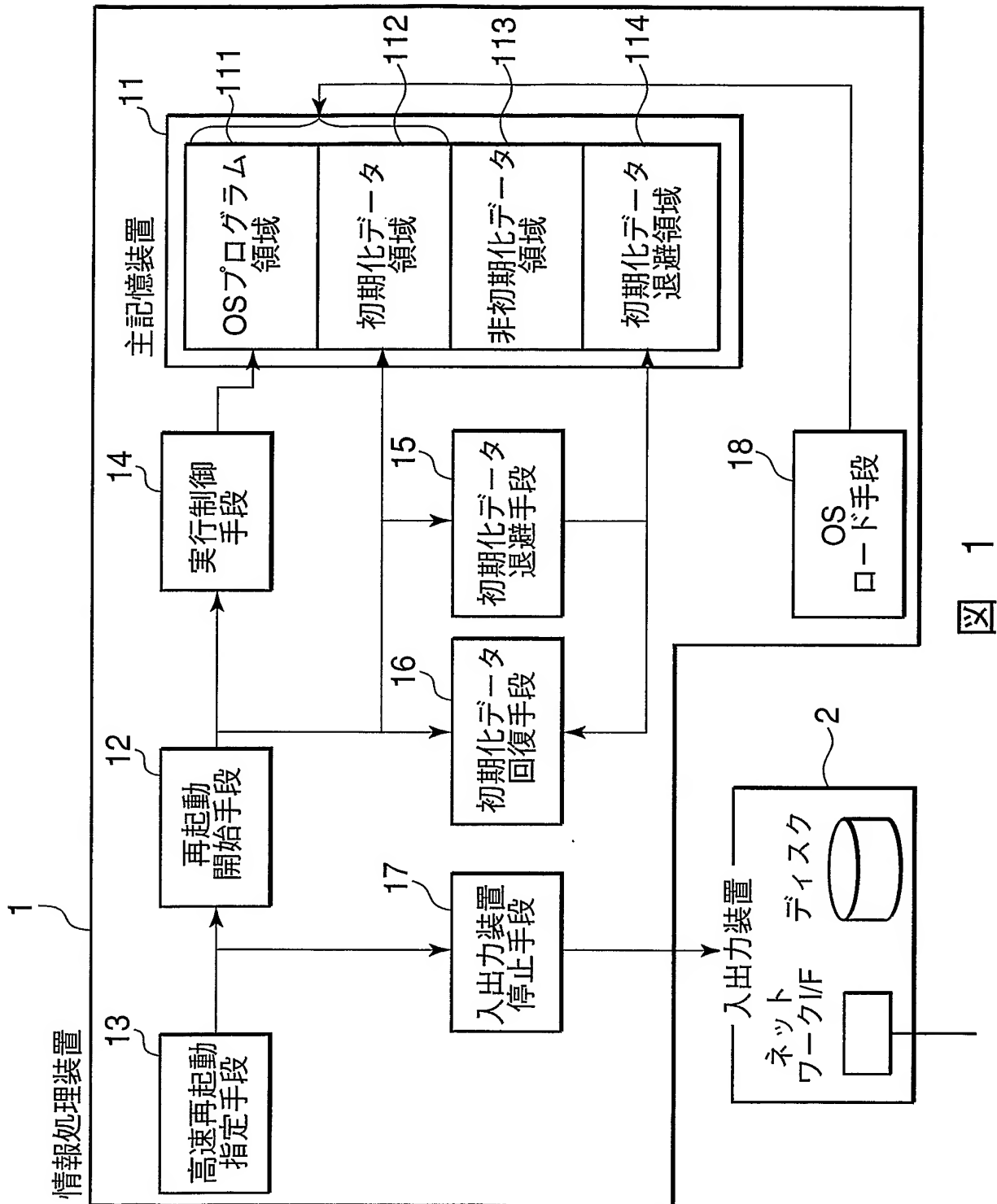


図 1

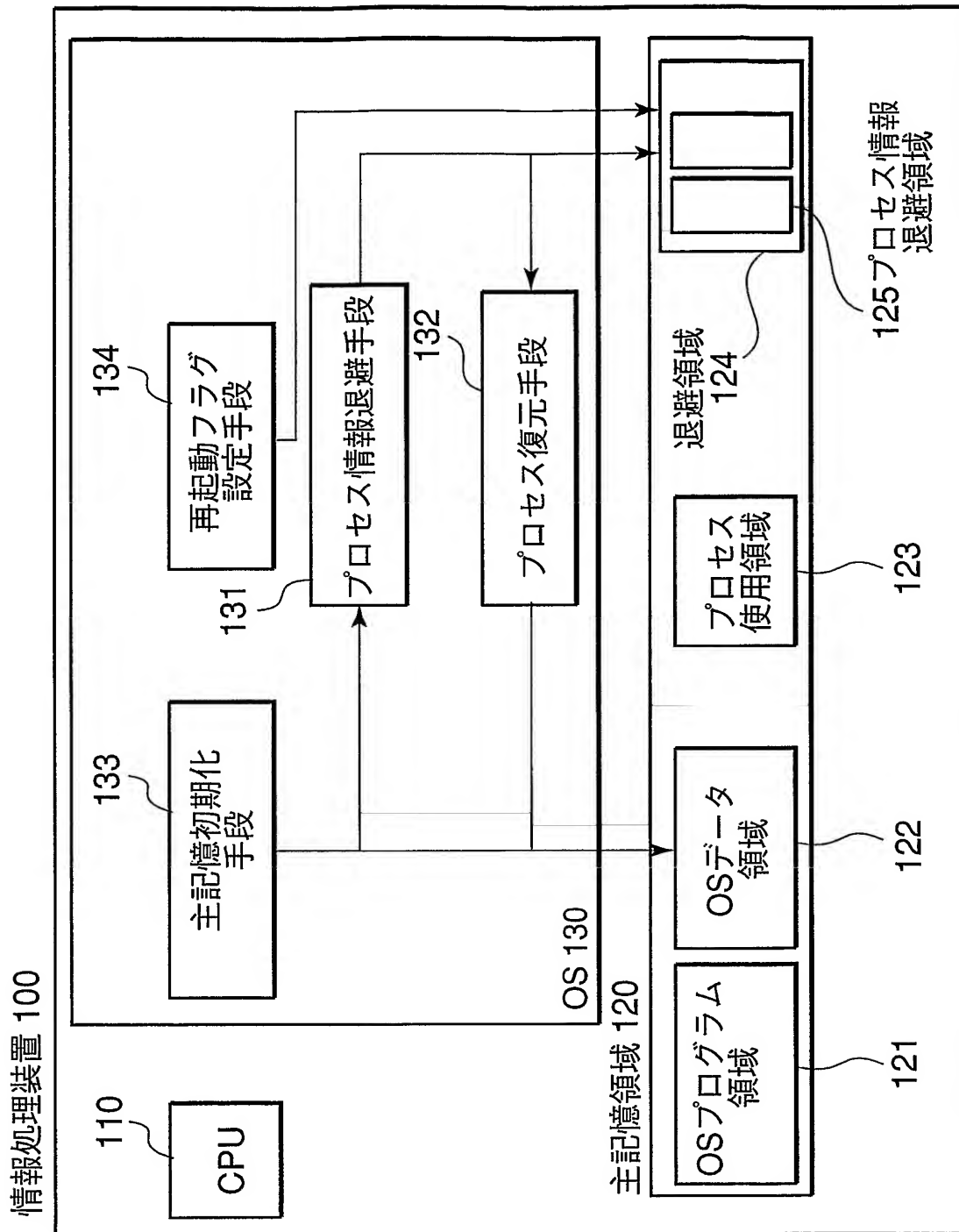


図 2

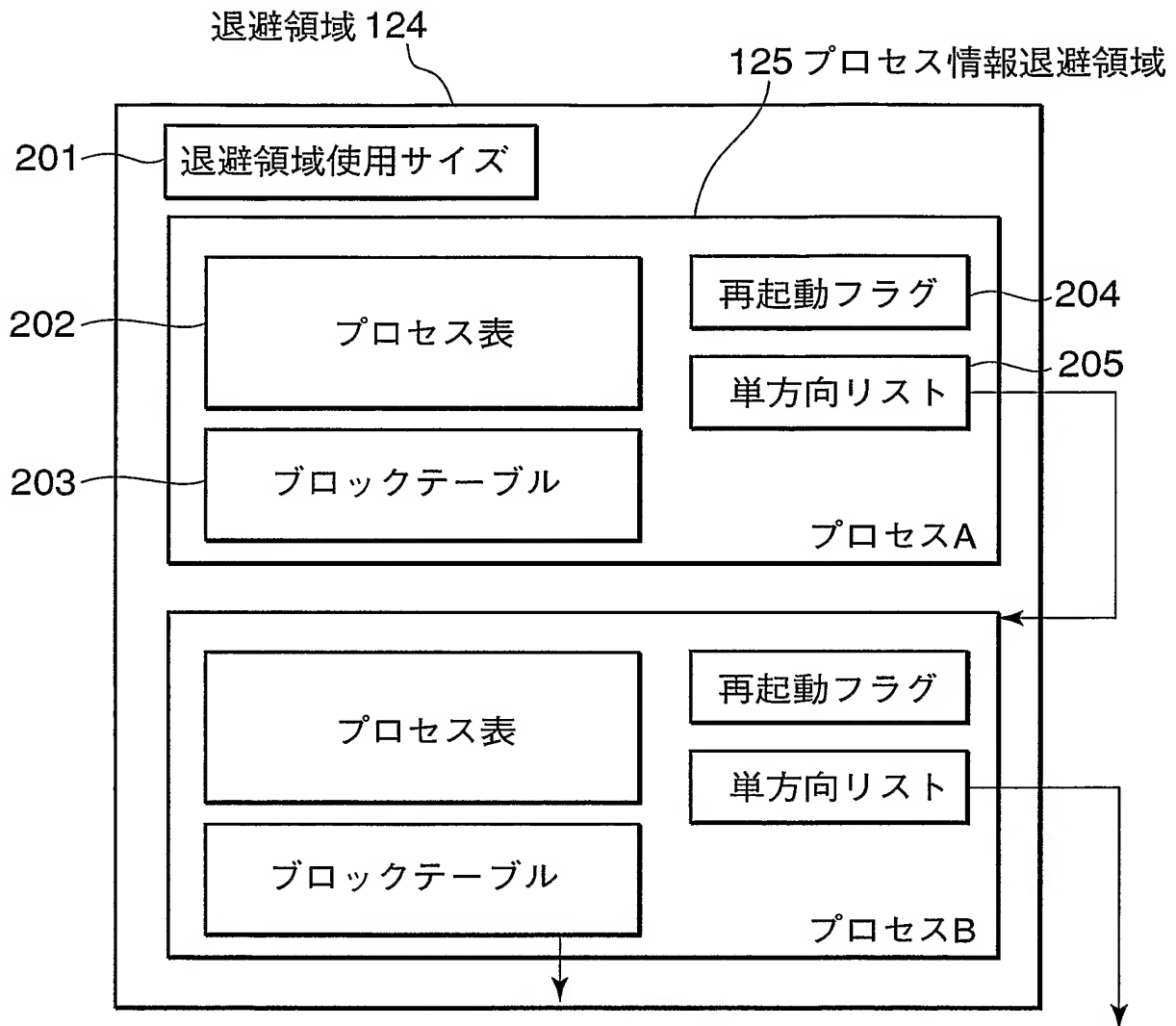


図 3

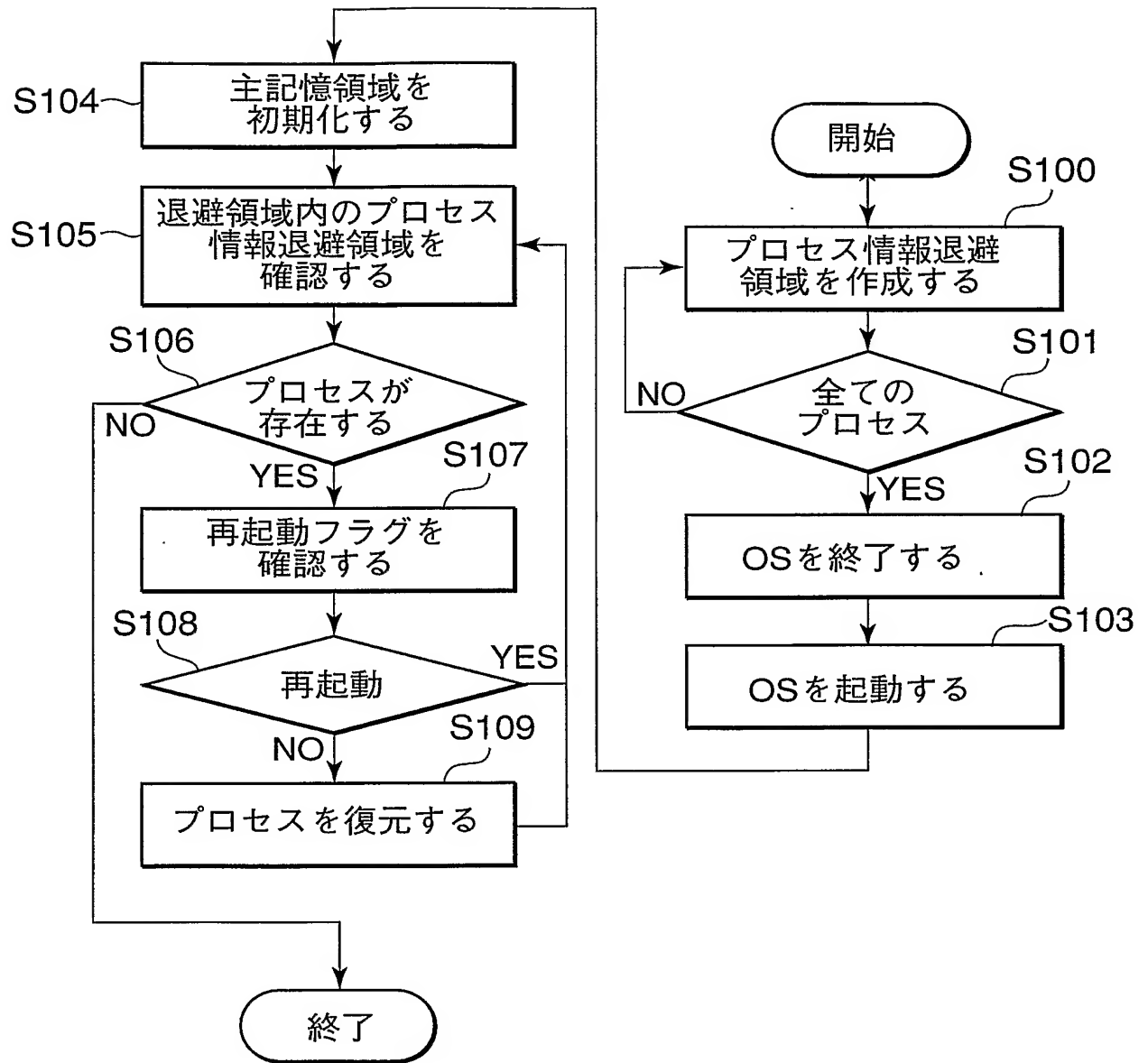
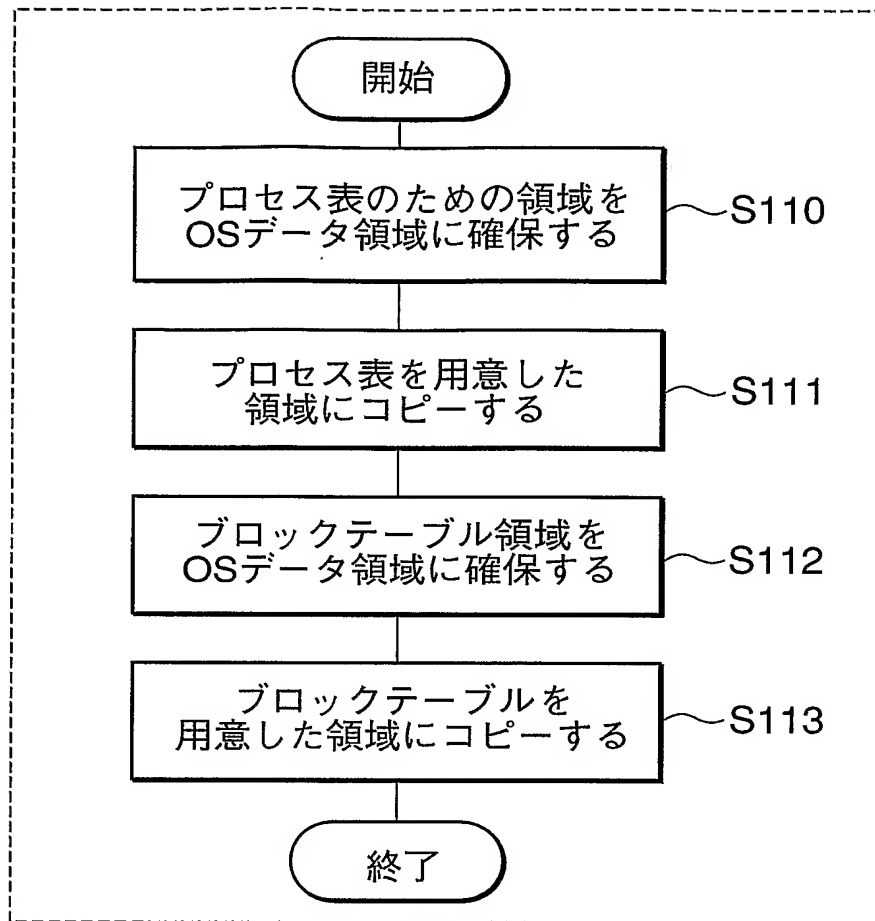
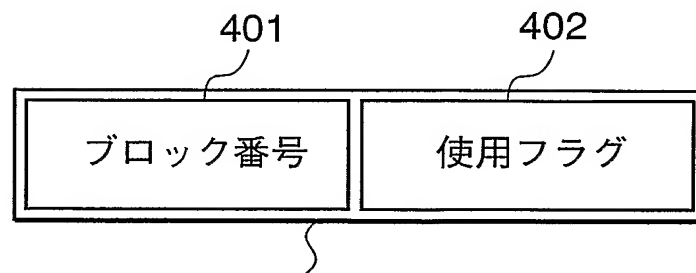


図 4A



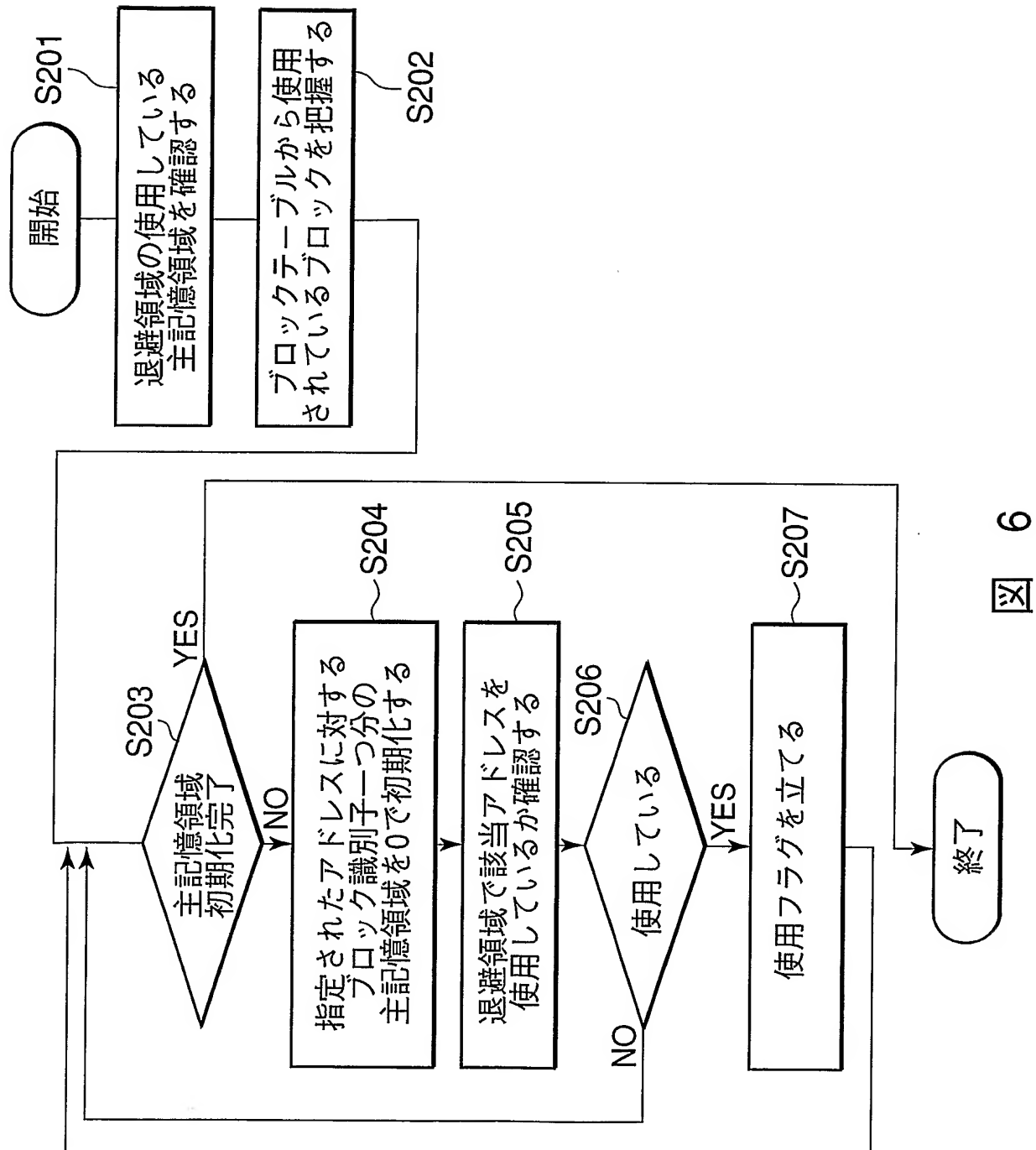
「プロセス復元」部分  
詳細フローチャート

図 4B



400 ブロック識別子

図 5



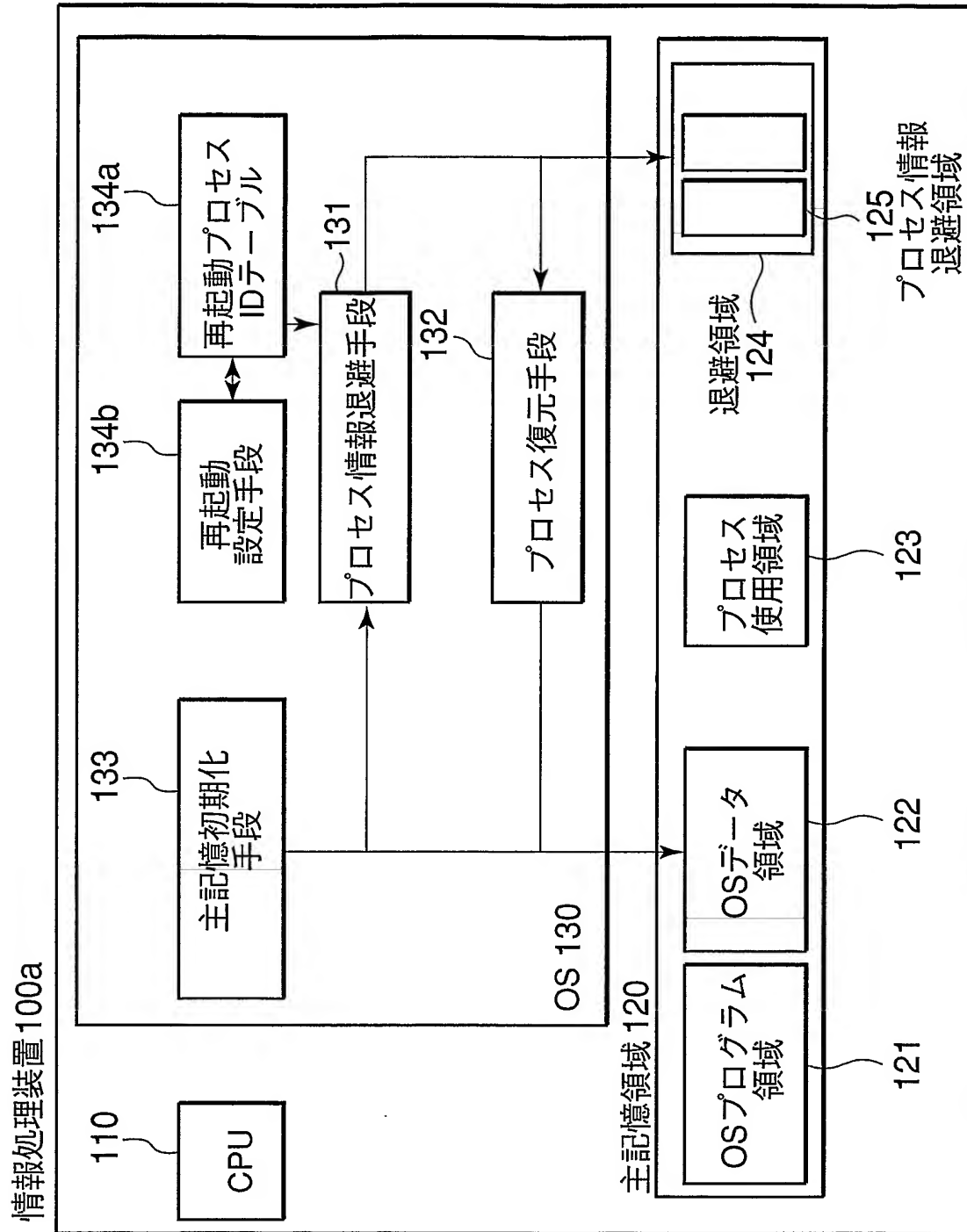


図 7

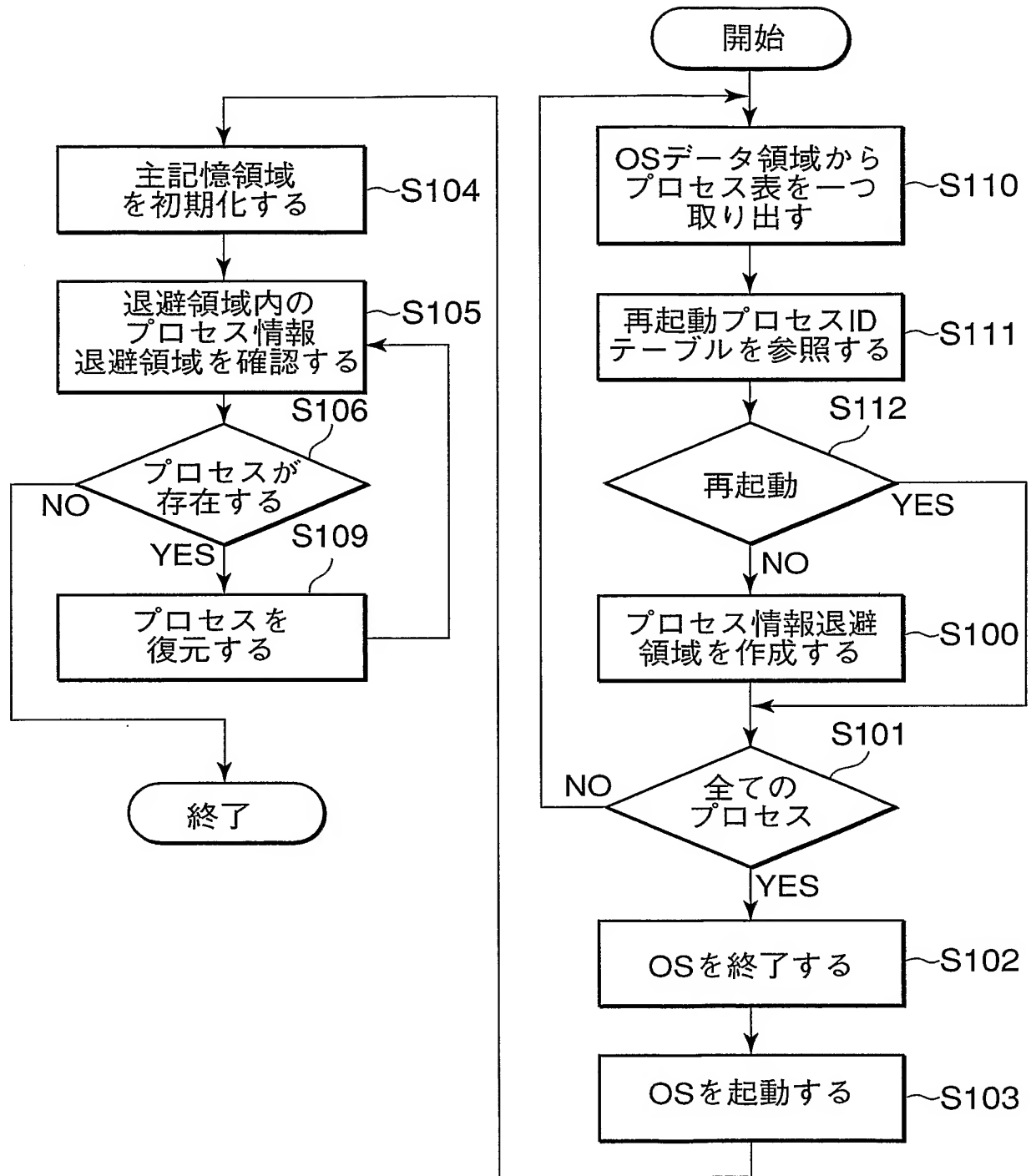


図 8

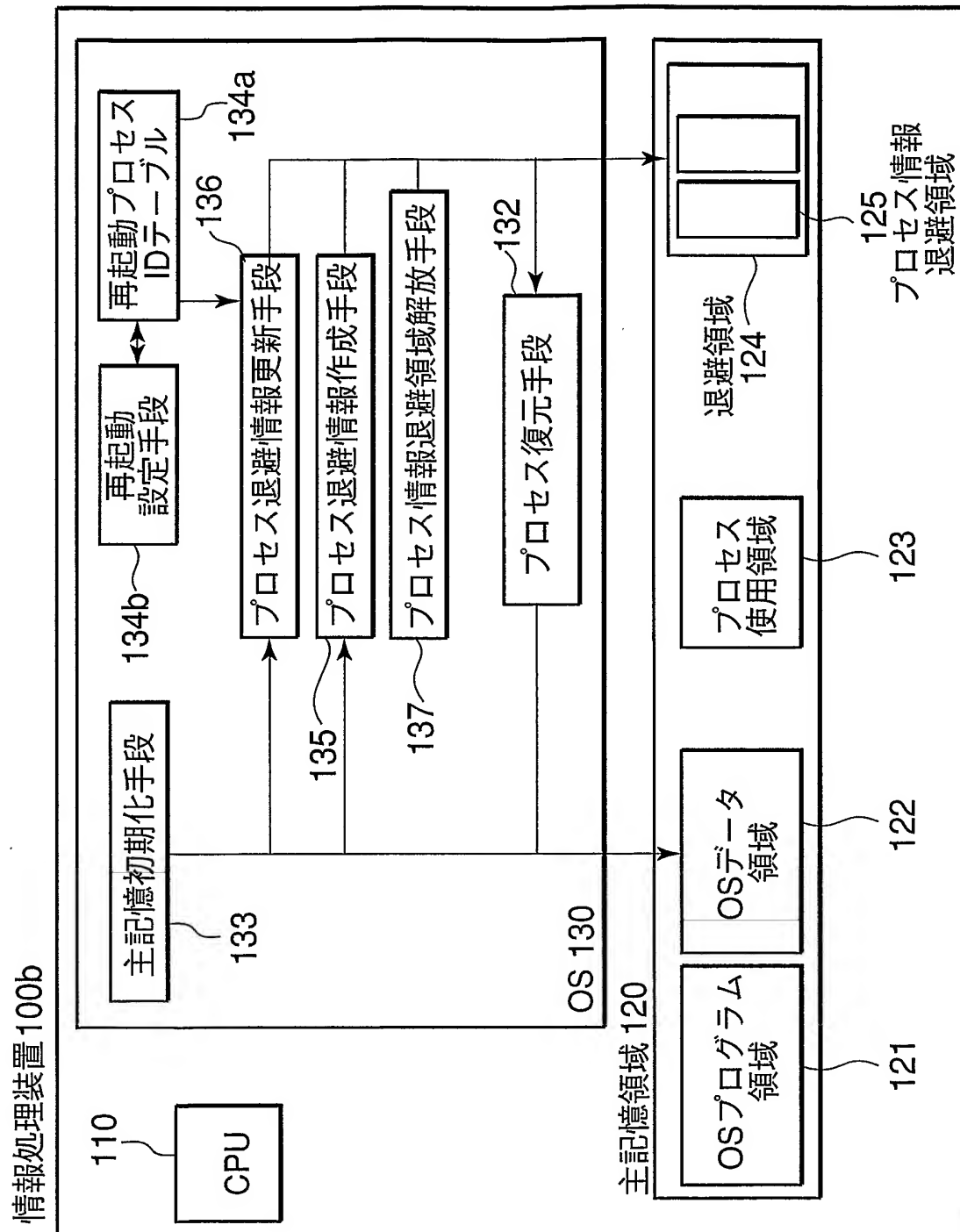


図 9

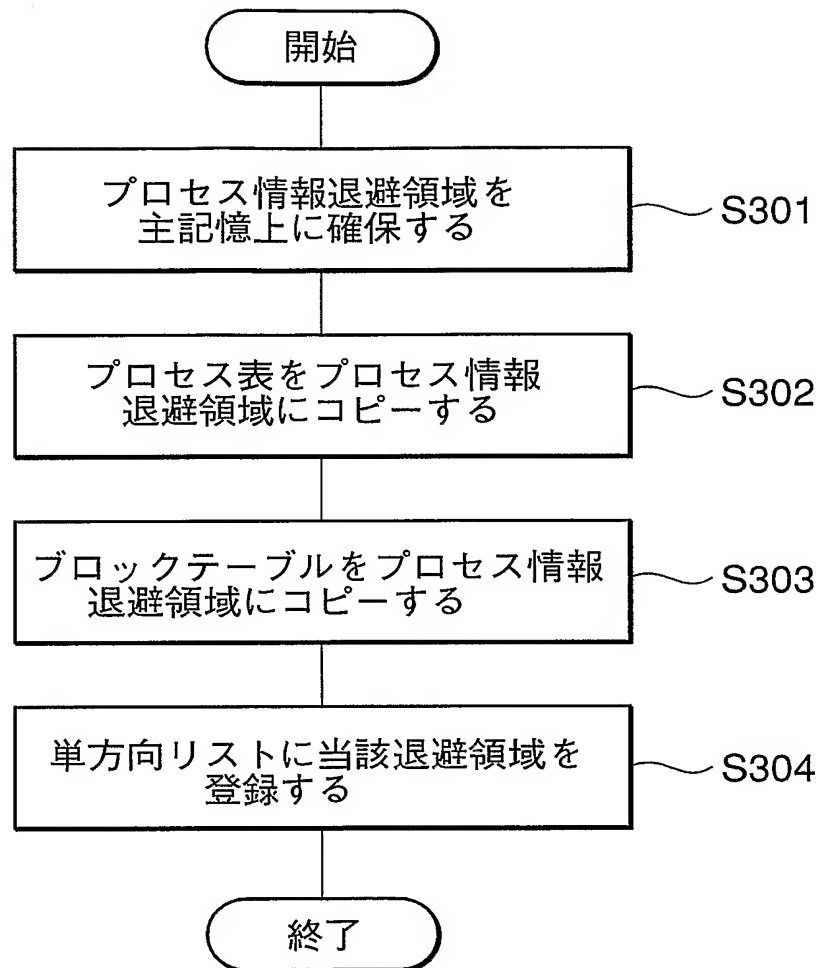


図 10

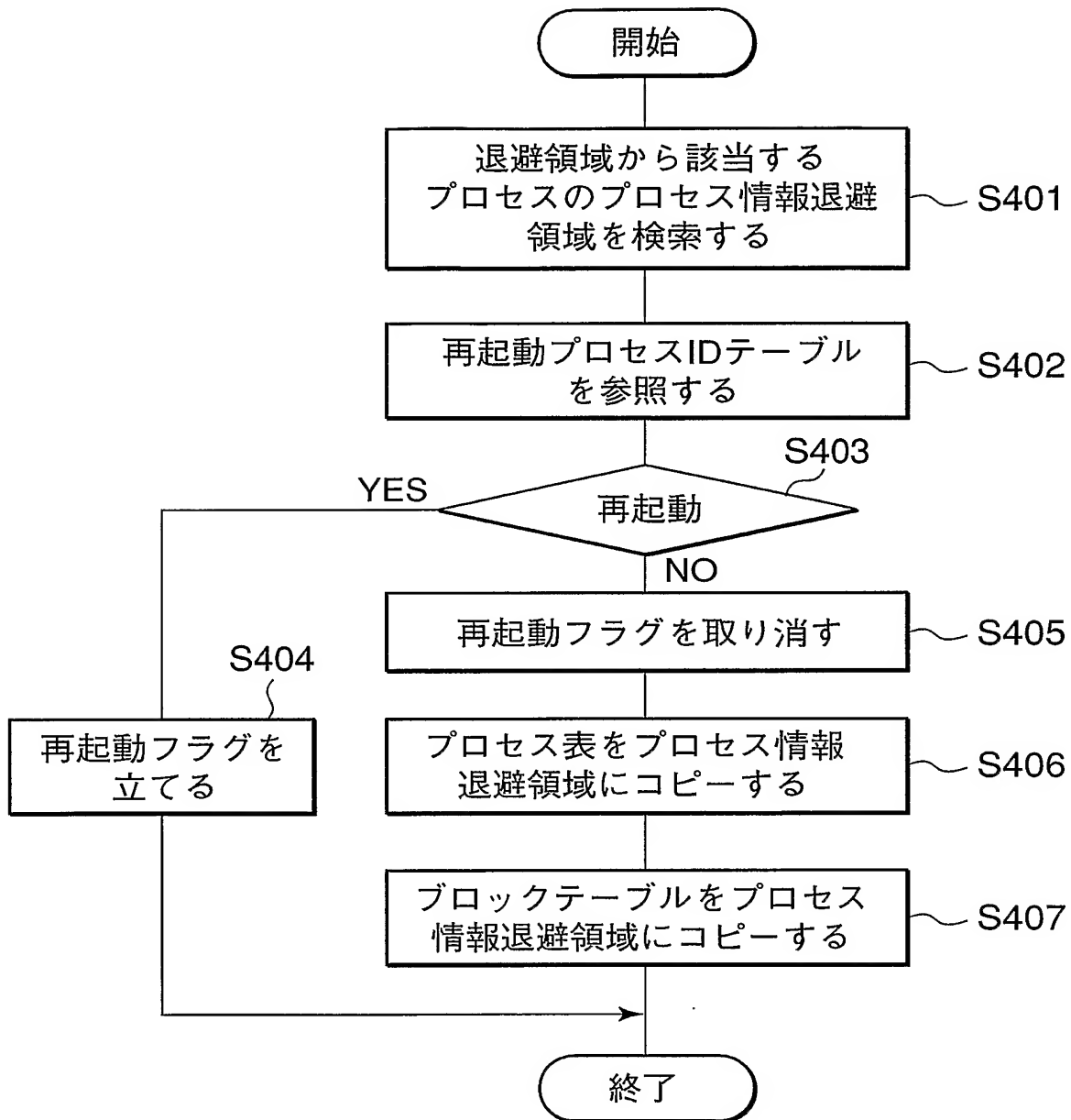


図 11

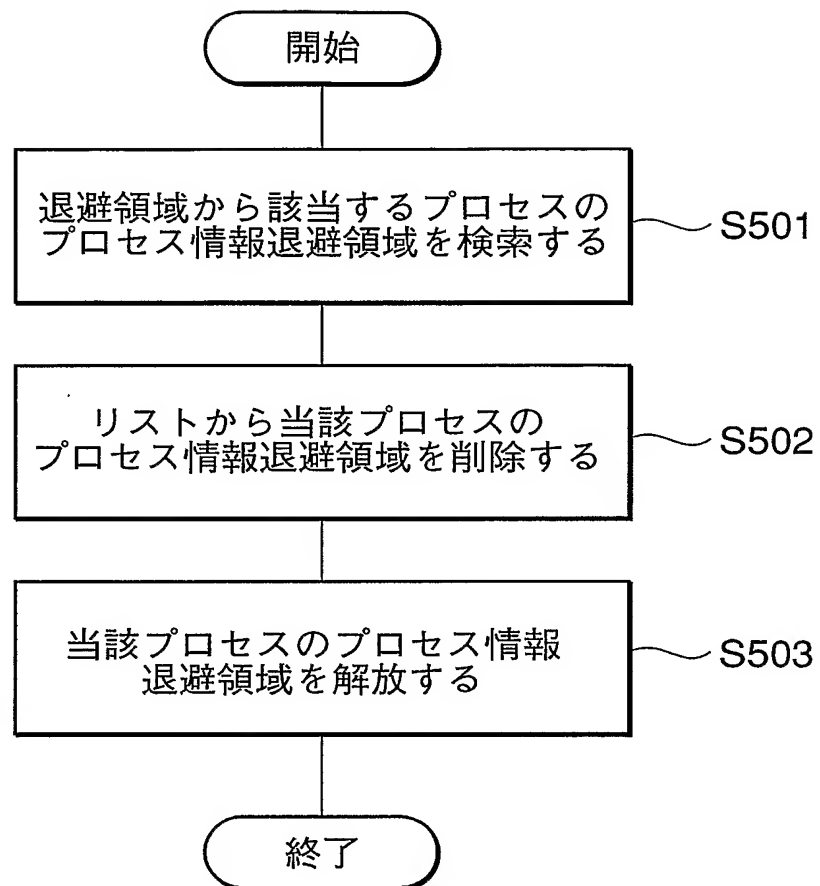


図 12

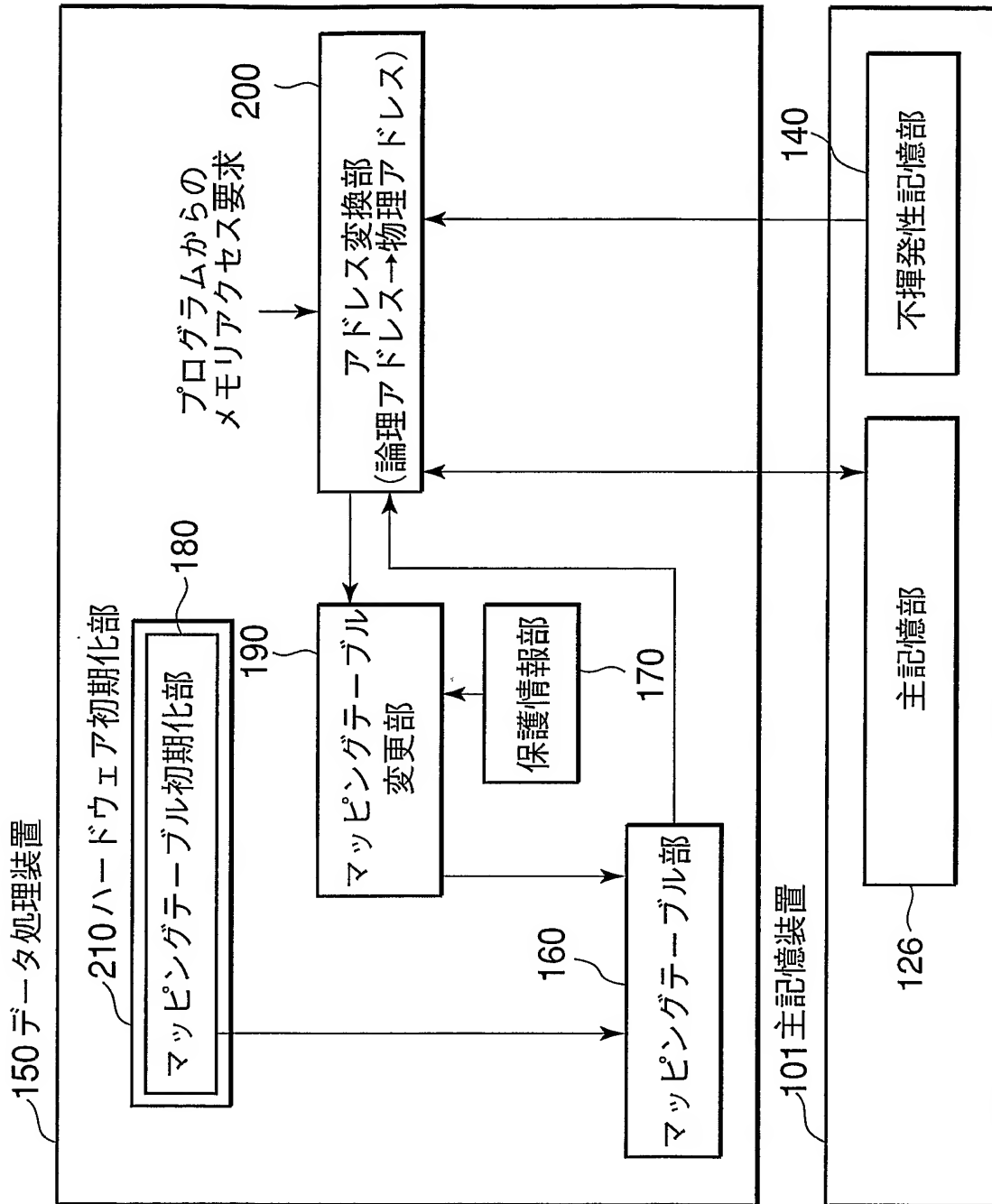


図 13

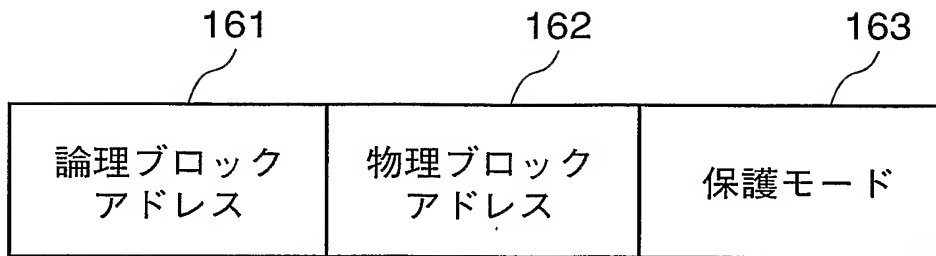


図 14

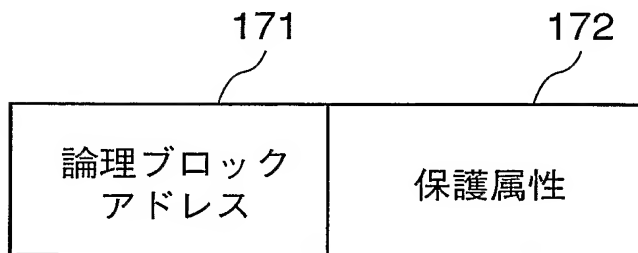


図 15

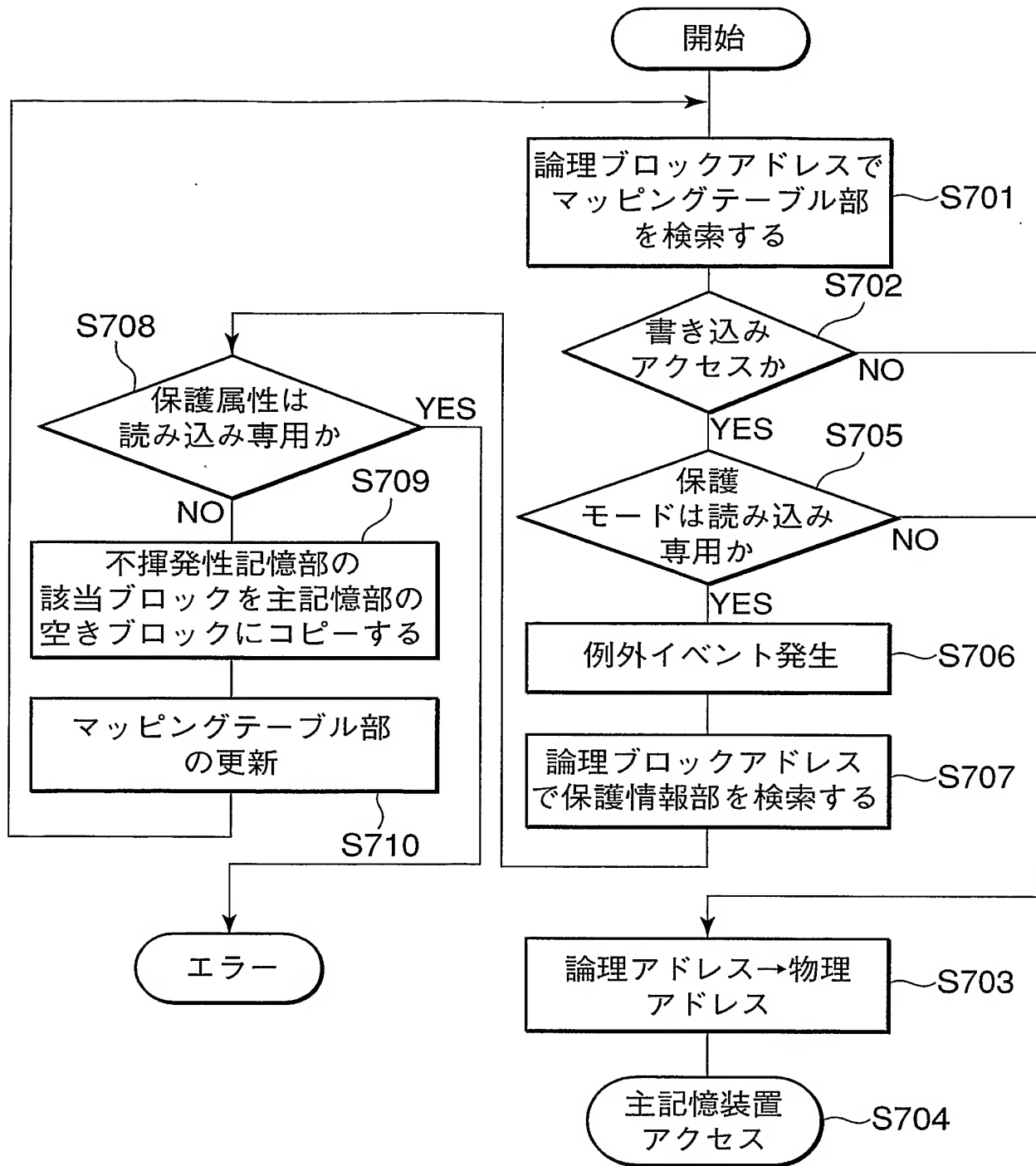


図 16

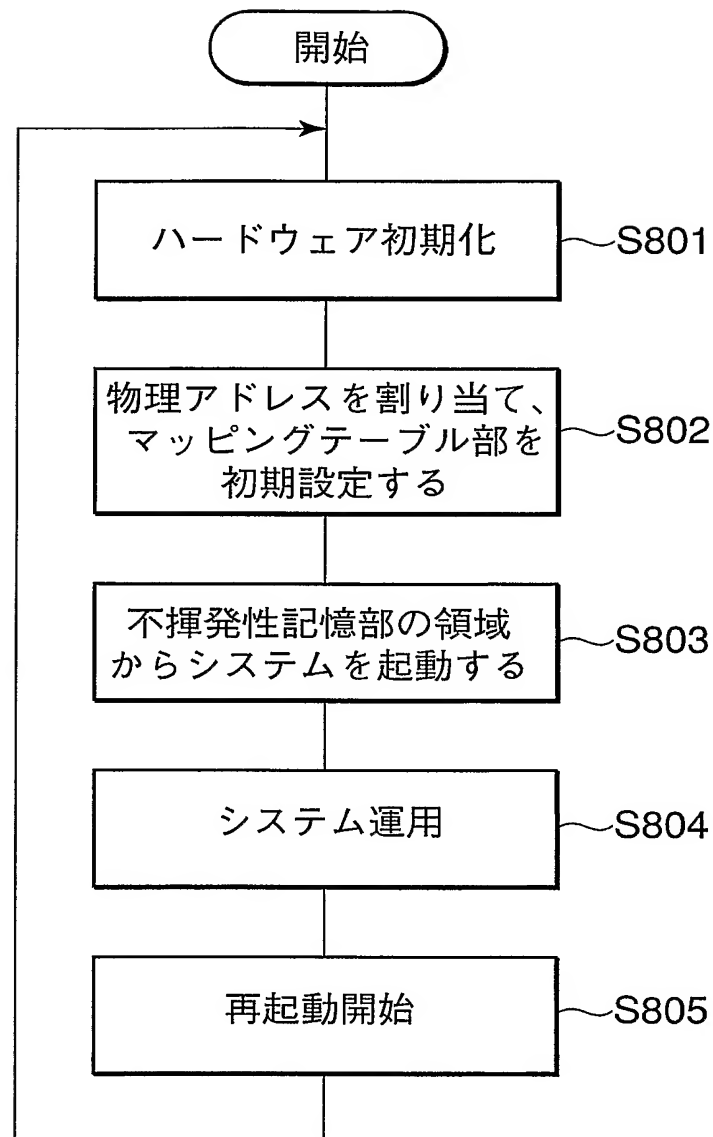


図 17

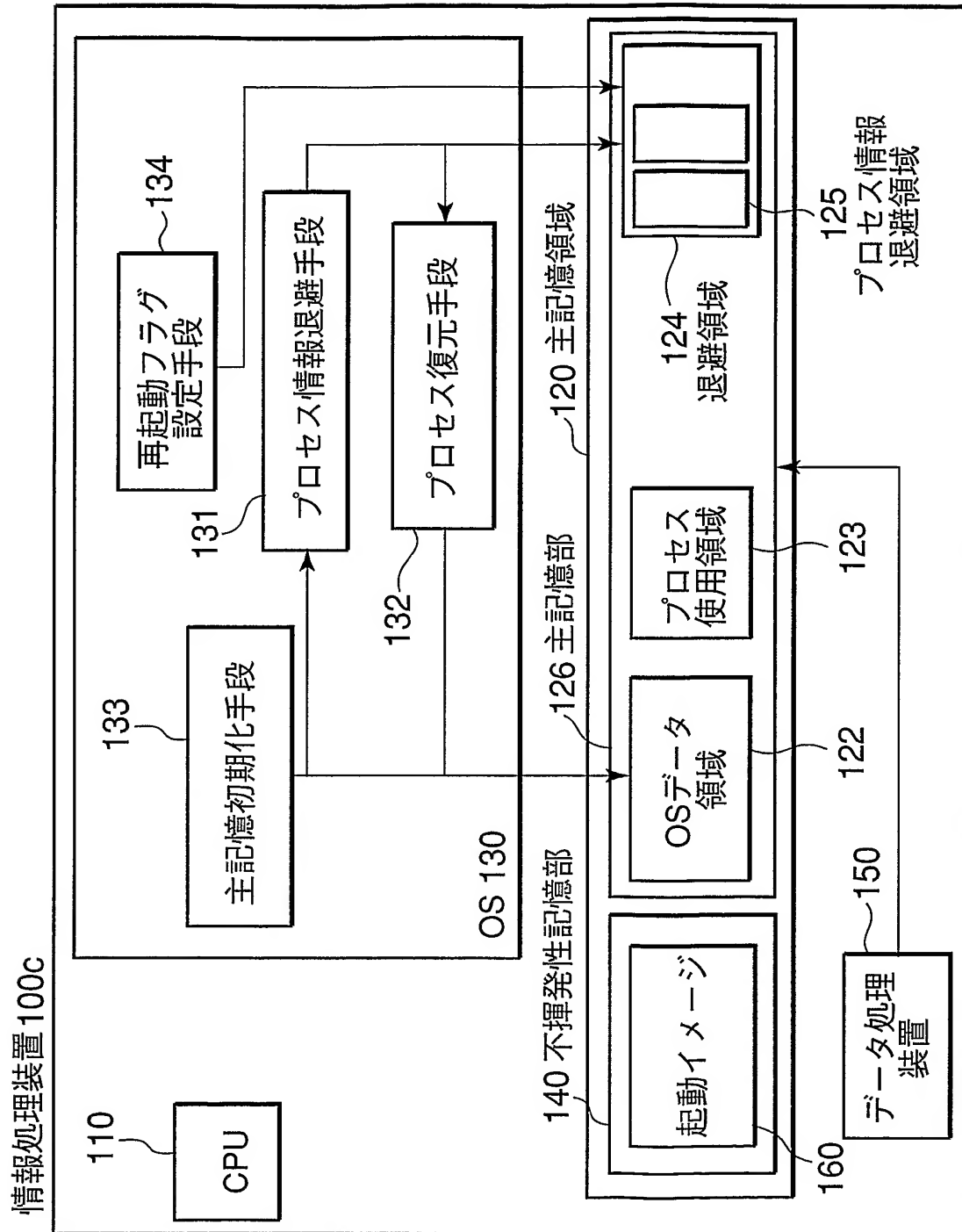


図 18

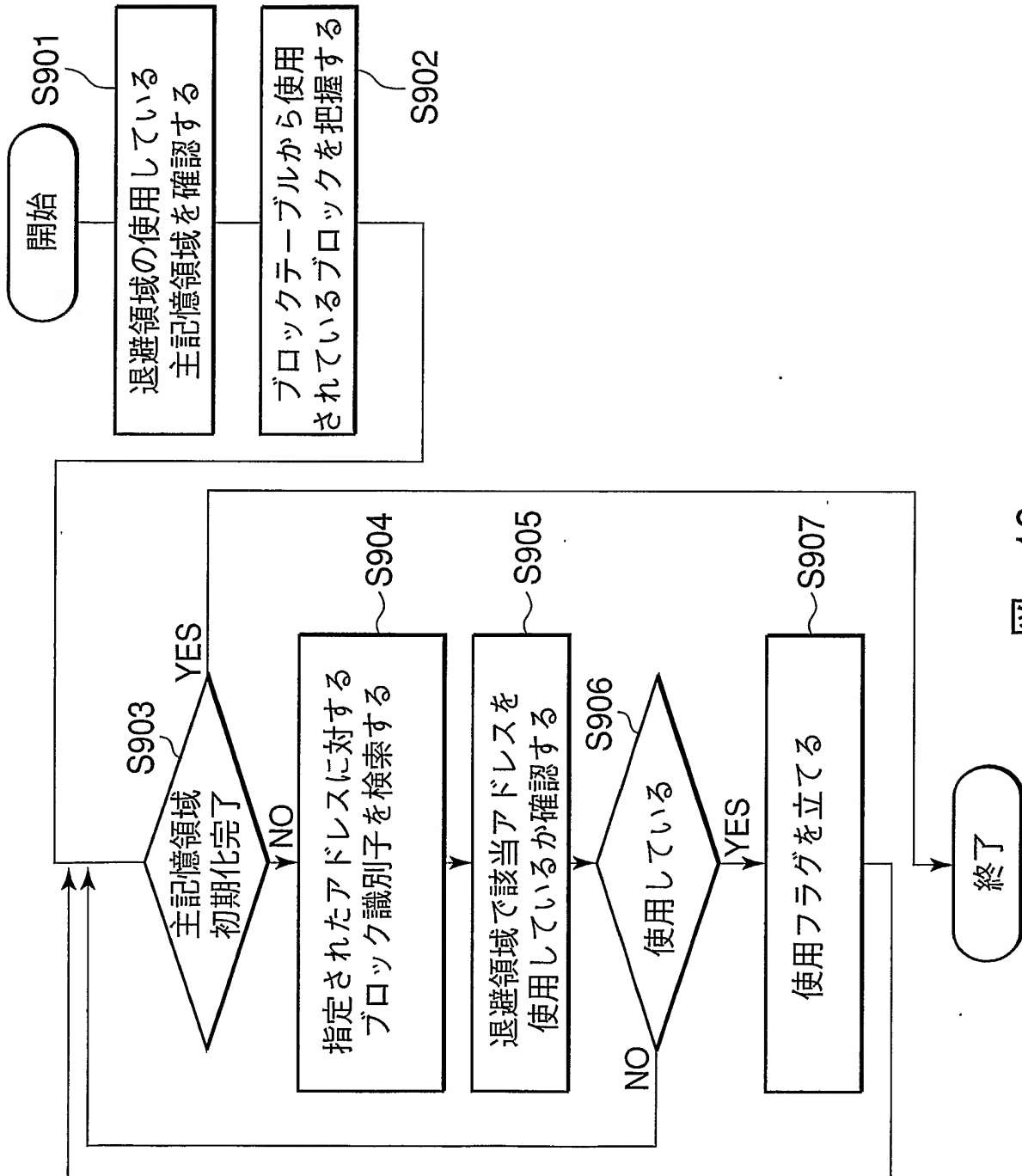


図 19

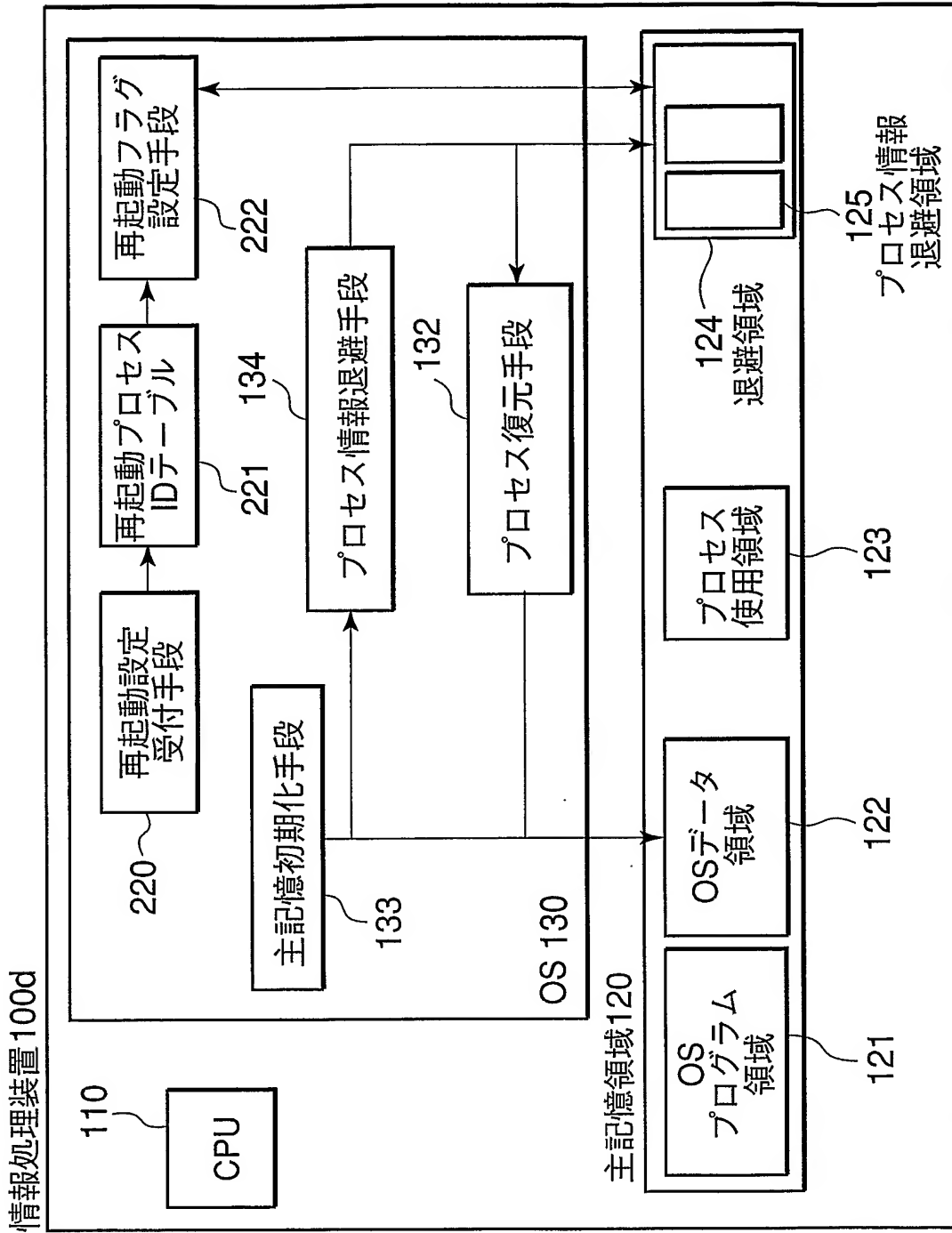


図 20

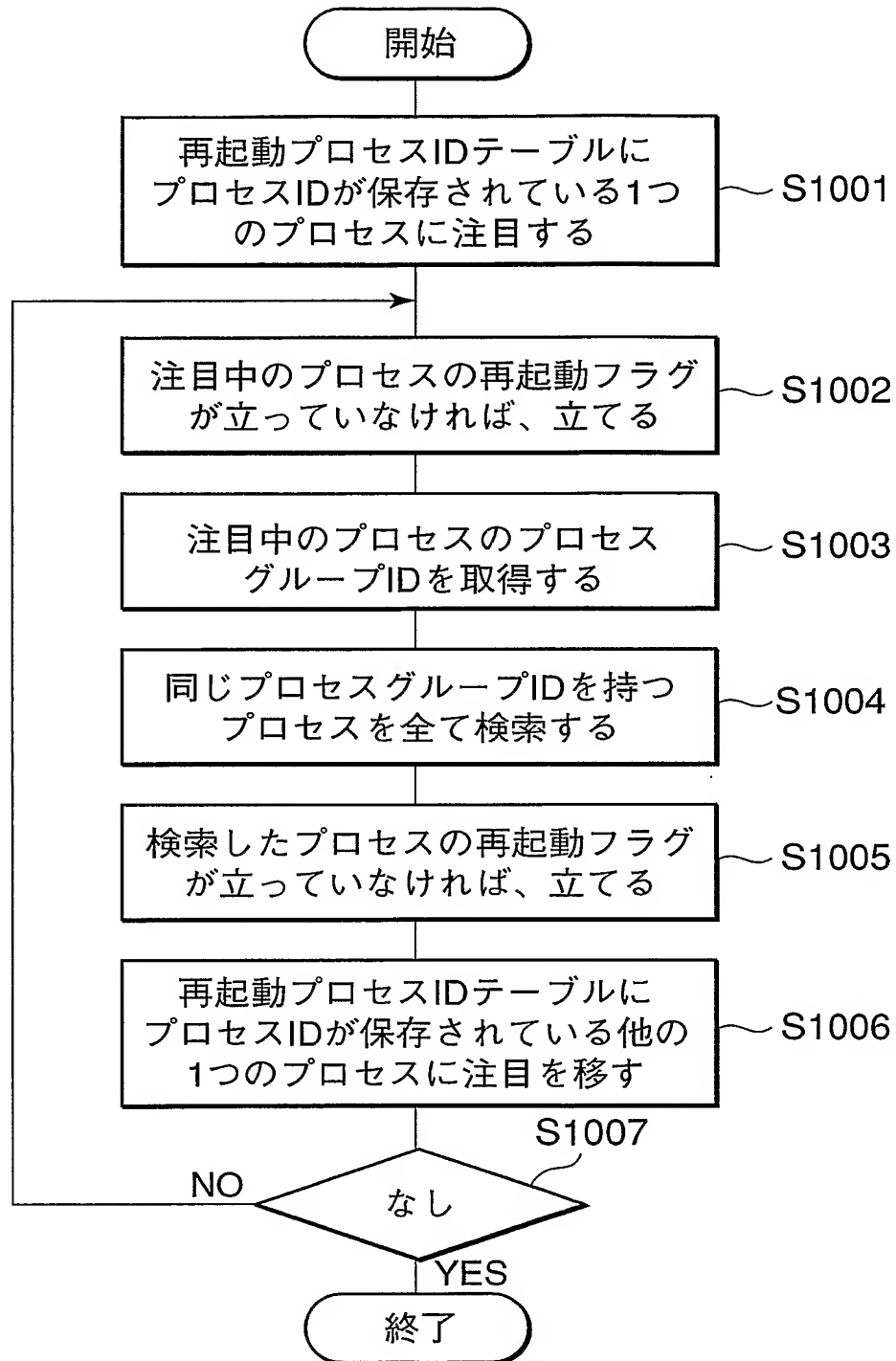


図 21

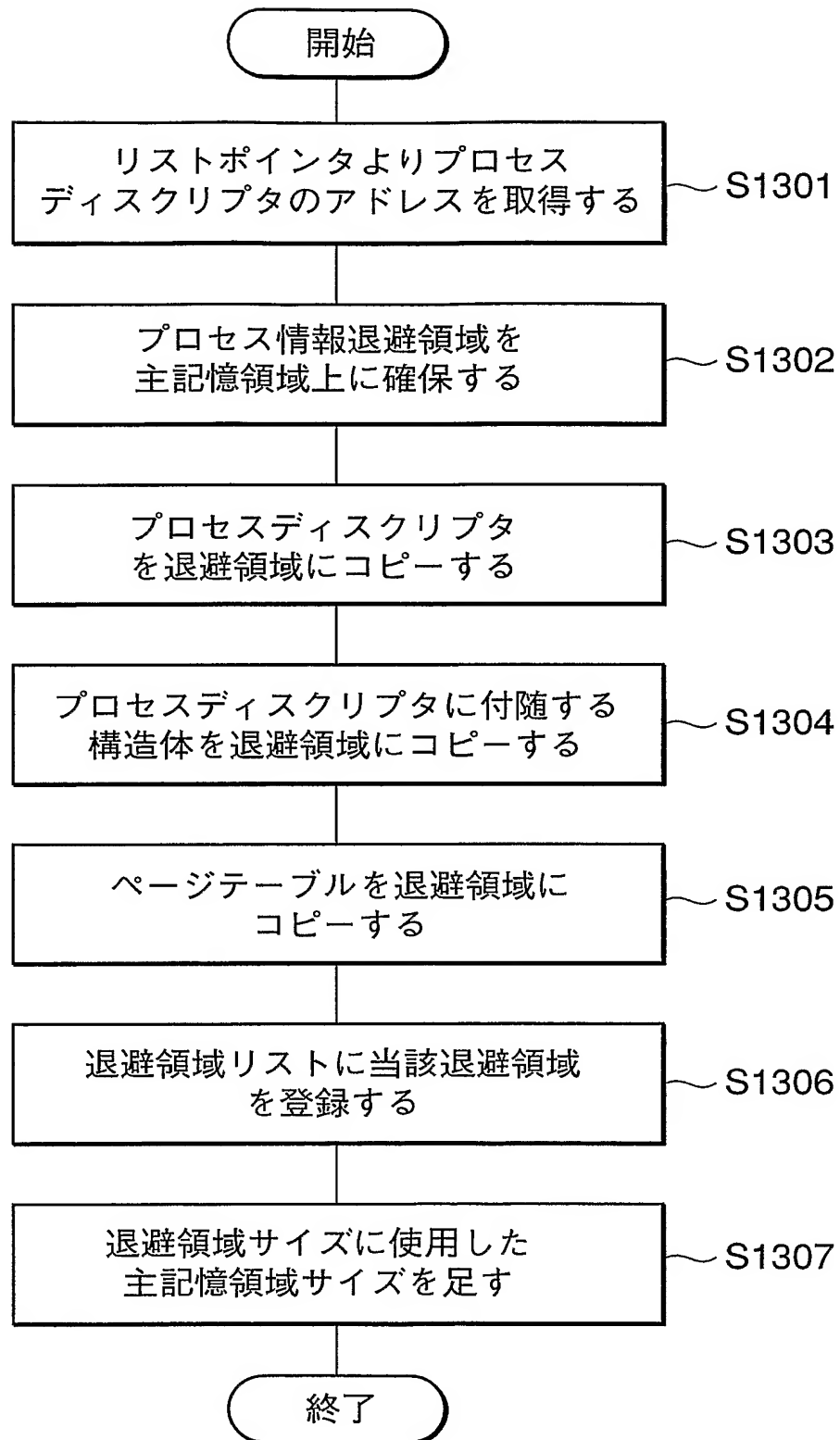


図 22

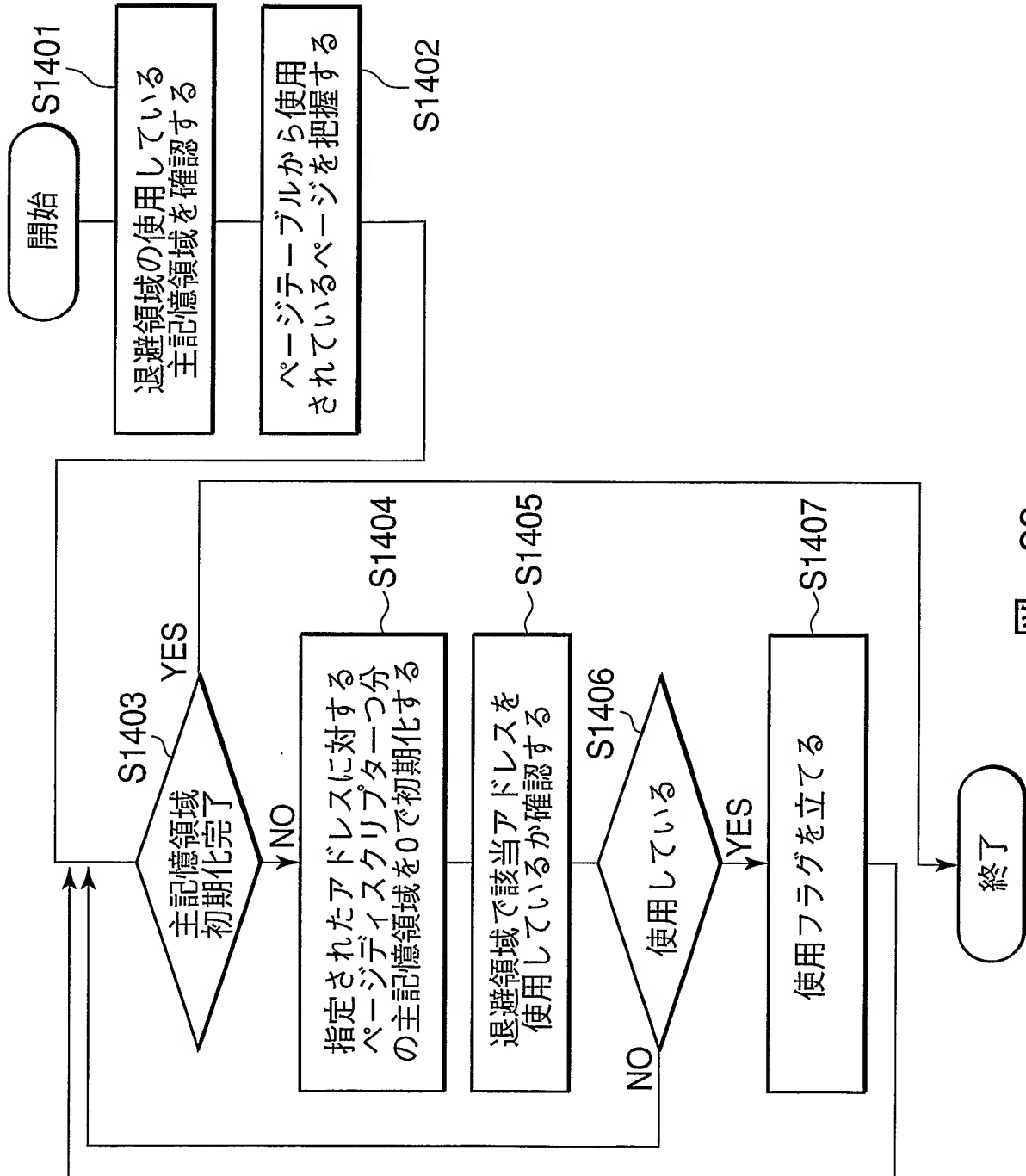


図 23

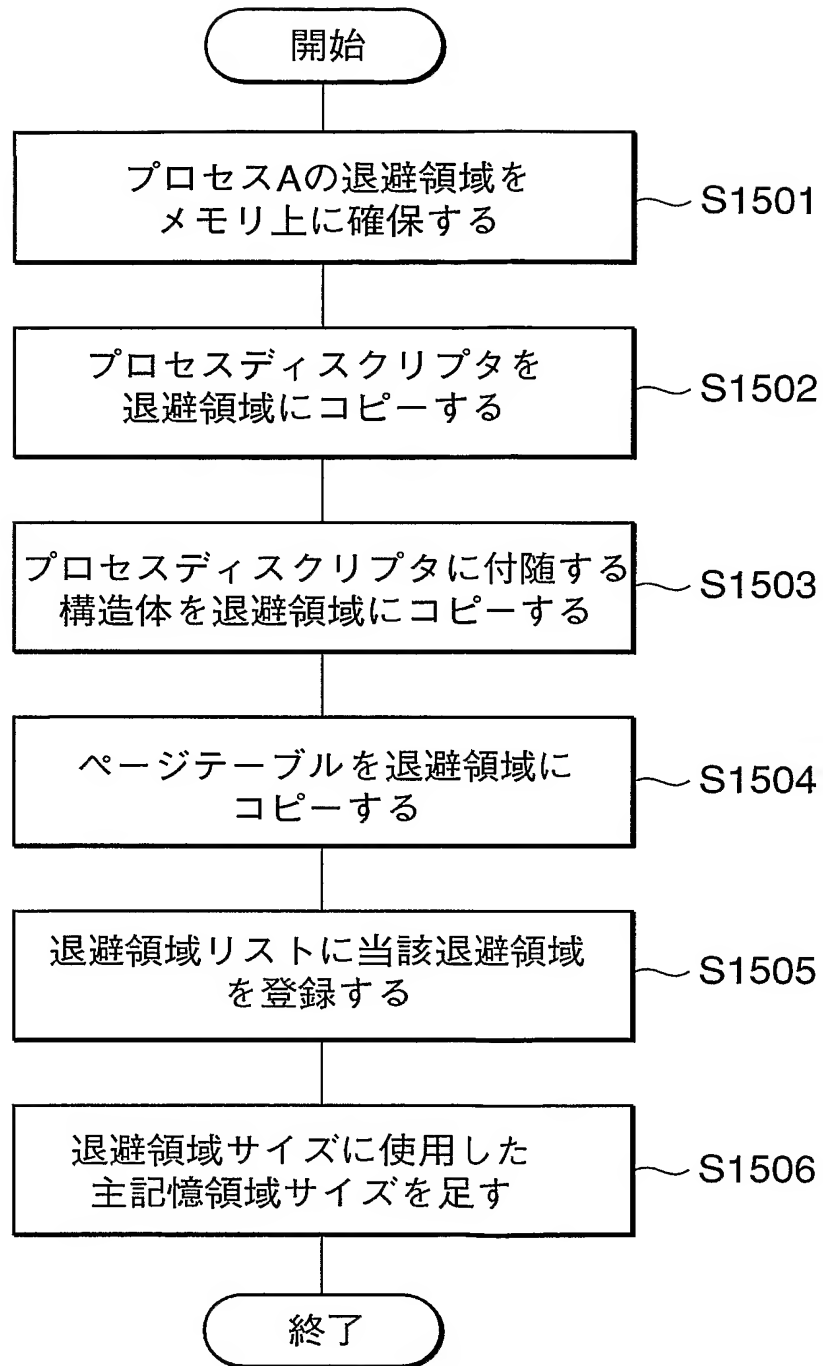


図 24

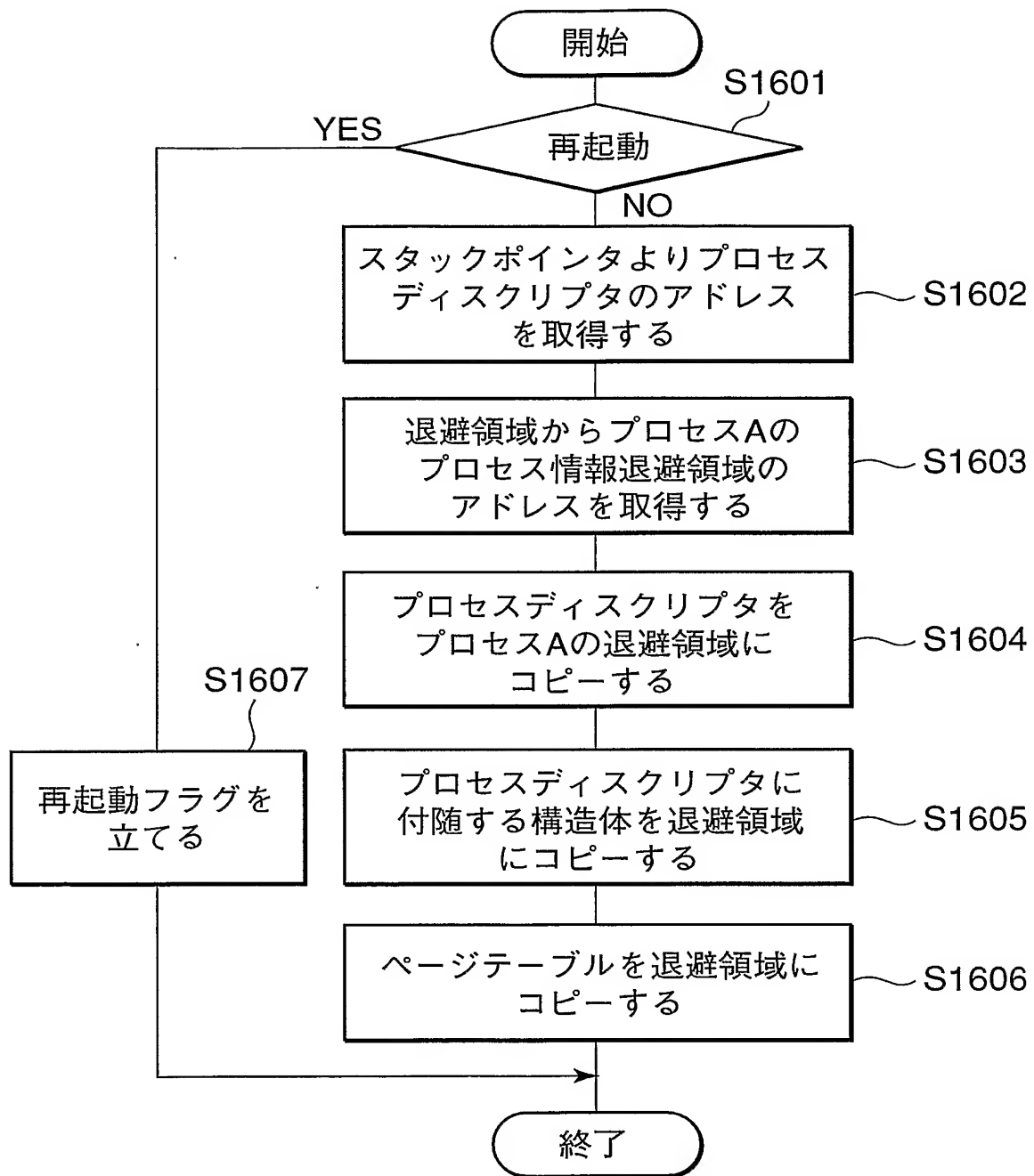


図 25

# INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2005/001559

A. CLASSIFICATION OF SUBJECT MATTER  
Int.Cl<sup>7</sup> G06F1/24, 9/445, 9/46

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)  
Int.Cl<sup>7</sup> G06F1/24, 9/445, 9/46

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched  
Jitsuyo Shinan Koho 1922-1996 Jitsuyo Shinan Toroku Koho 1996-2005  
Kokai Jitsuyo Shinan Koho 1971-2005 Toroku Jitsuyo Shinan Koho 1994-2005

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	JP 11-24936 A (Mitsubishi Electric Corp.), 29 January, 1999 (29.01.99), Full text (Family: none)	1-35
Y	JP 2002-259201 A (Hitachi, Ltd.), 13 September, 2002 (13.09.02), Full text (Family: none)	1-35
Y	JP 2002-82808 A (Seiko Epson Corp.), 22 March, 2002 (22.03.02), Full text (Family: none)	1-35



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A" document defining the general state of the art which is not considered to be of particular relevance	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E" earlier application or patent but published on or after the international filing date	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&" document member of the same patent family
"O" document referring to an oral disclosure, use, exhibition or other means	
"P" document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search  
14 April, 2005 (14.04.05)

Date of mailing of the international search report  
17 May, 2005 (17.05.05)

Name and mailing address of the ISA/  
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

## INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2005/001559

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	JP 5-27875 A (Canon Inc.), 05 February, 1993 (05.02.93), Full text (Family: none)	1-35
Y	JP 10-232796 A (Sony Corp.), 02 September, 1998 (02.09.98), Full text (Family: none)	1-35
Y	JP 11-345129 A (Lucent Technologies Inc.), 14 December, 1999 (14.12.99), Full text (Family: none)	1-35
Y	JP 2002-517034 A (Advanced Micro Devices Inc.), 11 June, 2002 (11.06.02), Full text & US 2002-13892 A & EP 1080407 A & WO 99/61981 A1	1-35
Y	JP 2000-35919 A (Victor Company Of Japan, Ltd.), 02 February, 2000 (02.02.00), Full text & US 2002-13892 A & EP 1080407 A & WO 99/61981 A1	2,5-7,11, 14-16,20,23, 25 4,13,22
A		
Y	JP 11-143764 A (Victor Company Of Japan, Ltd.), 28 May, 1999 (28.05.99), Full text (Family: none)	2,5-7,11, 14-16,20,23, 25 4,13,22
A		
A	JP 11-143764 A (Victor Company Of Japan, Ltd.), 28 May, 1999 (28.05.99), Full text (Family: none)	1-35

## A. 発明の属する分野の分類 (国際特許分類 (IPC))

Int.Cl.<sup>7</sup> G06F1/24, 9/445, 9/46

## B. 調査を行った分野

調査を行った最小限資料 (国際特許分類 (IPC))

Int.Cl.<sup>7</sup> G06F1/24, 9/445, 9/46

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報	1922-1996年
日本国公開実用新案公報	1971-2005年
日本国実用新案登録公報	1996-2005年
日本国登録実用新案公報	1994-2005年

国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)

## C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
Y	JP 11-24936 A (三菱電機株式会社) 1999. 01. 29, 全文 (ファミリーなし)	1-35
Y	JP 2002-259201 A (株式会社日立製作所) 2002. 09. 13, 全文 (ファミリーなし)	1-35
Y	JP 2002-82808 A (セイコーエプソン株式会社) 2002. 03. 22, 全文 (ファミリーなし)	1-35

☒ C欄の続きにも文献が列挙されている。☐ パテントファミリーに関する別紙を参照。

## \* 引用文献のカテゴリー

「A」 特に関連のある文献ではなく、一般的技術水準を示すもの  
「E」 国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの  
「L」 優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)  
「O」 口頭による開示、使用、展示等に言及する文献  
「P」 国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献

「T」 国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの

「X」 特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの

「Y」 特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの

「&amp;」 同一パテントファミリー文献

国際調査を完了した日

14. 04. 2005

国際調査報告の発送日

17.05.2005

国際調査機関の名称及びあて先

日本国特許庁 (ISA/J P)

郵便番号100-8915

東京都千代田区霞が関三丁目4番3号

特許庁審査官 (権限のある職員)

坂東 博司

電話番号 03-3581-1101 内線 3521

5E

4234

C (続き). 関連すると認められる文献		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
Y	JP 5-27875 A (キャノン株式会社) 1993. 02. 05, 全文 (ファミリーなし)	1-35
Y	JP 10-232796 A (ソニー株式会社) 1998. 09. 02, 全文 (ファミリーなし)	1-35
Y	JP 11-345129 A (ルーセント テクノロジーズ インコーポレイテッド) 1999. 12. 14, 全文 (ファミリーなし)	1-35
Y	JP 2002-517034 A (アドバンスト・マイクロ・デバイス・インコーポレイテッド) 2002. 06. 11, 全文 & US 2002-13892 A & EP 1080407 A & WO 99/61981 A1	1-35
Y A	JP 2000-35919 A (日本ビクター株式会社) 2000. 02. 02, 全文 & US 2002-13892 A & EP 1080407 A & WO 99/61981 A1	2, 5-7, 11, 14-16, 20, 23, 25 4, 13, 22
Y A	JP 11-143764 A (日本ビクター株式会社) 1999. 05. 28, 全文 (ファミリーなし)	2, 5-7, 11, 14-16, 20, 23, 25 4, 13, 22
A	JP 11-143764 A (日本ビクター) 1999. 05. 28, 全文 (ファミリーなし)	1-35